CLUBE

Julho/84

N.º 22

Coordenadores: MARIA IRENE • ALEXANDRE SOUSA • J. MAGALHÃES • AV. BOAVISTA, 832-2.º T. • TELEF. 65127 • 4100 PORTO

NESTE NÚMERO

Edição: Clube Z80

Fotocomposição: Fotomecânica Mabreu/Porto

Impressão: Ramos dos Santos & C.ª, Lda./Porto

Tiragem: 500 exemplares, Julho 1984

EDITORIAL

A revista CLUBE Z80, desde que se lançou, teve sempre como princípio fundamental não tratar a questão da Informática com uma concepção demasiadamente utópica e teórica, mas ser acessível, em termos práticos, ao utilizador mais inexperiente.
Nessa linha de ideias, julgamos estar a colocar realmente a máquina no seu devido lugar, articulando à prática a teoria necessária. Cremos que o conteúdo da revista cada vez mais se adequa às utilizações desejáveis e não a uma mera leitura que, depois, "fica na gaveta".
Contudo, parece-nos — e muitos sócios confirmam — que o esquema da revista está a tornar-se muito rotineiro. Há pouco tempo, Hugo Assumpção sugeria que variássemos um pouco, com novas abordagens, como p. ex., falar dos vários micros; Cibernética; Inteligencia Artificial; 5.ª Geração; Vendas de Computadores; CLUBE Z80 — Curva de Crescimento, Materiais Vendidos, Balanço Anual, Colaboradores, etc..
Assim, vamos tentar responder a estas e outras pretensões que surjam, alterando um pouco a "parte técnica" da revista, pois pensamos que a parte de programas serve perfeitamente os desejos dos utilizadores.
Ficamos também à espera das suas sugestões. Elas serão anotadas na nossa pauta e procuraremos, na medida do possível, viabilizá-las.

ALEXANDRE SOUSA
J. MAGALHÃES
MARIA IRENE

INTRODUÇÃO À LINGUAGEM MÁQUINA

ZX81/SPECTRUM

Autor: FERNANDO PRECES

(Cont. dos números anteriores)

PARTE III — COMO FUNCIONA O Z80 (Cont.)

4 — Programação em código máquina

Depois de uma introdução superficial sobre os componentes internos dum microcomputador e alguns detalhes de funcionamento dos seus principais órgãos, estamos em condições de iniciar um estudo mais profundo que nos relacione com a linguagem de acesso directo ao microprocessador Z80.
As instruções em linguagem máquina aparecem, quer no Manual do ZX81, quer no Manual do ZX Spectrum, escritas em Assembler, que emprega mnemónicas alfanuméricas para simbolizar operações e nomes dos vários órgãos do processador.
O Assembler é pois uma linguagem que utiliza as mnemónicas como idioma intermédio entre o Homem e a máquina.
Tal como as linhas de instruções BASIC, também o códido máquina é arrumado de igual forma na RAM, com a diferença que importa salientar do BASIC ter um endereço inicial invariável, e o C.M. poder ser colocado em qualquer endereço disponível da memória, que se encontre **acima da RAMTOP.**
Porquê acima da RAMTOP? Pode interrogar-se o leitor menos esclarecido.
Porque abaixo dessa zona, as faixas de memória ocupadas pelas variáveis e pelas pilhas máquina (ver o manual) **se deslocam a cada instante ao longo da RAM,** conforme as solicitações da programação. O código máquina necessita, por razões que veremos mais tarde, de estar depositado em **localizações permanentes,** que serão obtidas depois de se **interditar** uma zona da memória à acção do monitor.
No ZX81, para de **consignar** a RAMTOP, uma vez determinado o endereço respectivo extensão, (este depende da extensão do bloco ou da rotina em C.M. que queremos introduzir), é necessário calcular os valores do (Low Byte Address) e do (High Byte Address), com os quais devemos carregar a **variável do sistema** (RAMTOP), endereços 16388/9.

Sendo X o endereço, teremos:

X = Y * 256 + Z e calcula-se:
M = (X/256)
Y = (High byte) = INT m
Z = (Low byte) = ((m − Y) * 256)
POKE 16 388 , Z
POKE 16 389 , Y e **NEW**

No Spectrum, é bastante mais simples.

Apenas:

CLEAR X e **ENTER**

Assim, um grupo de bytes acima da RAMTOP é totalmente ignorado pelas operações rotineiras da programação BASIC, e somente o comando POKE o pode alterar. NEW apenas limpa a memória abaixo da RAMTOP.
Para além de números, alfabeto e sinais ortográficos, o Basic do ZX81 possui uns 70 comandos utilizáveis a partir do teclado (o Spectrum tem uns 100), que podem desencadear através do monitor determinadas funções do Z80.
Dada a grande diferença existente entre as duas máquinas, já do conhecimento dos nossos leitores, teremos de admitir que essa, na sua maior parte, se situa a nível das ROMs (a do ZX81 com 8 K bytes e do Spectrum com 16 K bytes de extensão).
No ZX81, os endereços compreendidos entre 8.193 e 16.383, na realidade, não existem. A informação que recebemos com um PEEK n (endereços citados) dá-nos uma **imagem** da ROM, **repetindo os conteúdos** dos endereços entre 0 e 8.192, devido a um arranjo de ligações eléctricas, na base do chip.
Voltaremos mais tarde a falar sobre esse **espaço vago** e como o poderemos utilizar.
Para elaborar um programa monitor da envergadura destas ROMs, são necessárias algumas dezenas de técnicos altamente especializados nos campos das matemáticas e da

electrónica digital. Outros requisitos não relacionados com a técnica também entram em linha de conta, pois é normalmente o fabricante do computador quem determina o que quer vender, mediante os produtos lançados no mercado pela concorrência, ou o tipo de utilizador que tem por objectivo atingir.
É esta última, a razão fundamental porque o Basic destas máquinas Sinclair é tão vigiado (elas vêm preparadas para o utilizador inexperiente). A única forma de contornar essa limitação, que torna o Basic-Sinclair extraordinariamente lento, é utilizar em linguagem máquina a alta velocidade do seu processador, aproveitando as potencialidades proporcionadas pela manipulação das 700 instruções do seu Assembler.
Pegando nos manuais respectivos (ZX81, páginas 187 a 193 e Spectrum, páginas 171 a 178) podemos verificar que 252 dessas instruções são formadas por um único byte (todas as que se encontram na primeira coluna) e as restantes por 2 bytes.
Algumas dessas instruções exigem um complemento numérico que, associados, formam composições com 2, 3 e 4 bytes de extensão.
O tempo gasto na execução duma instrução Assembler é dependente dessa extensão, que está relacionada com o número de funções a cumprir, tais como:

a) Tempo de comutação do apontador e recolha de uma cópia do conteúdo do endereço n. Esta acção é constante para todo o tipo de instrução.

b) Câmbio de valores entre registros indigitados. Um grande número de instruções que envolvem um único byte e que provocam a carga de registro com conteúdo de outro, ou as que provocam o câmbio de conteúdos entre registros, são de execução muito rápida (soma dos tempos a e b).

c) Recolha de cópia do conteúdo do endereço (n + seguintes) para o complemento numérico da instrução (1 ou 2 bytes). Este tipo de instrução é executada num tempo (a + c) ou (a + 2 c).

d) Transferência dum conteúdo de registro para um endereço X da RAM. (Soma dos tempos (a e d) para 1 byte ou (a + 2 d) para 2 bytes).

e) Intervenção da ALU (unidade de aritmética e lógica). O tempo de execução de algumas instruções é agravado por esta intervenção, que é muito curta pelo facto de esta unidade apenas efectuar pequenas operações binárias, mas que não pode deixar de ser referida.

A Unidade de Control uniformiza rigorosamente esses tempos, servindo-se de uma **batuta** (o ciclo máquina), que é sincronizado por impulsos introduzidos na entrada **Clock** do processador.

4.1 — O Clock e o Ciclo Máquina

Um circuito comandado a cristal de quartzo (tal como nos relógios electrónicos) fornece os impulsos de sincronismo que comandam toda a actividade do Z80. Este elabora, a partir deles, 2 ciclos máquina de diferente duração.
O primeiro (M1) contém 4 ciclos T (ciclo T é igual a um período do Clock) e o segundo (M2) contém 3 ciclos T.
Chama-se **Ciclo de Instrução** ao tempo de recolha e execução duma instrução que, como já vimos, pode ter uma maior ou menor duração, conforme a sua complexidade, sendo a extensão medida em ciclos T.

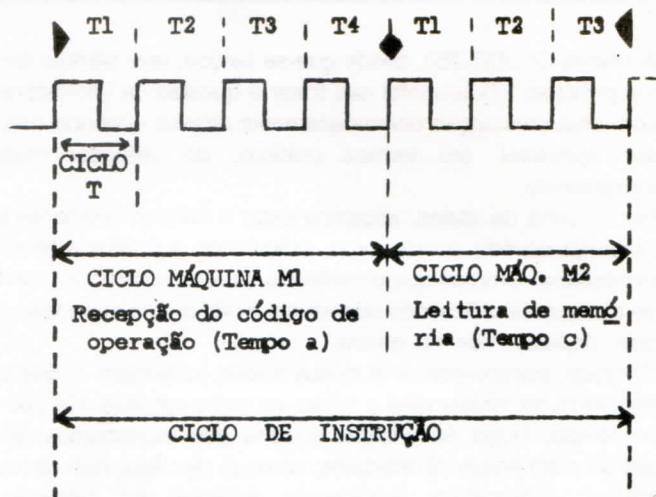



FIG. 4.1 — Exemplo de um Ciclo de Instrução para uma mnemónica que ordena a carga dum registro simples com um conteúdo do endereço (n + 1).

O ciclo T tem uma duração diferente em cada máquina (para o ZX81, 307,6 nanosegundos e para o Spectrum 285,7 nanosegundos), a que corresponde para o ZX81 uma frequência de trabalho de 3,25 megaciclos e para o Spectrum uma de 3,5 megaciclos.
No exemplo apresentado na figura 4.1, o tempo necessário para a execução dessa instrução é no ZX81 de $7 * 307{,}6$ nanosegundos (2,1532 microsegundos $\rightarrow 2{,}1532 * 10^{-6}$ do segundo) e no Spectrum de $7 * 285{,}7$ nanosegundos (1,9999 microsegundos $\rightarrow 1{,}9999 * 10^{-6}$ do segundo).

4.2 — O salto para uma rotina máquina

Tal como um GO SUB produz um salto para a Linha n, aonde se encontra a subrotina que se pretende executar e o RETURN provoca o regresso imediato à instrução seguinte do ponto de salto, também o comando USR X produz um salto para determinada rotina máquina que se inicia no endereço X e a mnemónica RET provoca o retorno incondicional ao Basic, à instrução seguinte do ponto de salto.
O comando USR, cujo argumento X é sempre um endereço, obriga o Z80 a **interromper** a sequência do programa monitor e a percorrer uma outra com **início em x** (1.° endereço da Rotina máquina) situada ou na RAM (C.M. do programador) ou na ROM (C.M. do monitor).
Note-se que num salto USR X, para uma rotina memorizada na RAM, **não podemos** utilizar o registro IY e o alternativo HL, sem primeiro memorizar os seus conteúdos, que terão de ser restituídos aos mesmos no instante de retorno (antes de RET).
Se tal não acontecer, ou o programa se imobiliza por estar desactivada uma interrupção de máscara (o apontador IY perdeu o seu endereço base), ou se dá um colapso total e tudo desaparece (o alternativo HL perde o ponto de reentrada no monitor).
Em qualquer dos casos, o utilizador terá de desligar a máquina e recomeçar o trabalho.
As formas mais usuais de se empregar nas 2 máquinas o comando USR, são:

PRINT USR X — Que executa o C.M. e regressa escrevendo no ecran o valor decimal contido no registo BC, na posição PRINT do ficheiro, endereçada pela variável DF CC (23 684/5, no Spectrum) e variável DF CC (16 398/9, no ZX81).

RANDOMIZE USR X ou RAND USR X — Que activa o gerador de números aleatórios antes do salto.

LET A = USR X — Que utiliza uma variável adicional, equivalente à função da rotina.

A mnemónica RET (código 201), obriga o Z80 a voltar de imediato à sequência interrompida quando do salto USR X.

Ensaio:

PROGRAMA 0 — Utilização do comando USR e da instrução RET.

ZX81

Para 1 K RAM RAMTOP em 17151, 2 K RAM em 18075, 16 K RAM em 32410.
Sugerimos que, para os ensaios a efectuar, o utilizador do ZX81, consoante a extensão de memória da sua máquina escolha para a RAMTOP, um dos endereços acima indicados.

```
                                        mnemónicas
10  SLOW
20  POKE (RAMTOP + 1), 201              RET
30  STOP
40  LET A = USR (RAMTOP + 1)
50  PRINT AT 12,7 ; "DE VOLTA AO BASIC"
```

SPECTRUM

Para 16 KAM RAMTOP em 31999 e 48 K RAM em 64999

```
                                        mnemónicas
10  CLEAR (RAMTOP)
20  POKE (RAMTOP + 1), 201              RET
30  STOP
40  LET A = USR (RAMTOP + 1)
50  PRINT AT 12,7 ; "DE VOLTA AO BASIC"
```

Depois de escrever o programa, → RUN. O único código desta Rotina vai ser memorizado e o programa pára na linha 30. A seguir → RUN 40 e o Z80 salta para o endereço indicado, lê o código de retorno e volta ao Basic (linha 50).

4.3 — As mnemónicas do Z80

GRUPO I

A instrução NOP (não operativo), nada mais faz de que provocar um tempo de atraso de 4 ciclos T na recolha da instrução seguinte, pois não obriga o processador a produzir qualquer trabalho. Nenhum registro ou flag é afectado.
A instrução é utilizada em C. M. como **enchimento** para produzir certos efeitos ou rectificar erros.

Exemplo:

Produzir um compasso de espera, para atrasar o movimento de uma figura, para a leitura duma legenda, apagamento de uma ou mais instruções que pretendamos eliminar, ou ainda, para reservar espaços dentro duma rotina com o propósito de introduzir mais tarde algumas instruções que de início não tenham sido previstas.

mnemónicas	Código de operações	N.° Bytes	N.° ciclos M	N.° ciclos T
NOP	0	1	1	4

Ensaio:

PROGRAMA 1 — Instrução NOP. Para a RAMTOP, proceder como no ensaio anterior.

ZX81

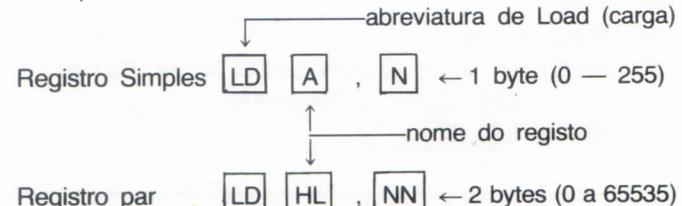
```
                                        mnemónicas
10  CLS
20  FOR A = 1 TO 99
30  POKE (RAMTOP + A), 0                NOP
40  NEXT A
50  POKE (RAMTOP + 100), 201            RET
60  STOP
70  LET A = USR (RAMTOP + 1)
80  PRINT AT 12,7 ; "DE VOLTA AO BASIC"
```

Para o Spectrum, substituir a linha 10 por CLEAR RAMTOP: CLS.
Introduzir RUN, aguarde o STOP e depois RUN 70.

Comentários:

A rotina é formada por 99 NOPs, e 1 RET.

GRUPO 2 — Este grupo é constituído por todas as instruções que representam a carga de registros com um ou dois bytes de dados, copiados dos endereços seguintes.

Exemplo:

Carga de registros simples

mnemónicas	códigos
LD A , N	62
LD H , N	38
LD L , N	46
LD B , N	6
LD C , N	14
LD D , N	22
LD E , N	30

Tempo de execução destas instruções:

N.° Bytes	N.° ciclos M	N.° ciclos T
2	2	7

Carga de registros pares

mnemónicas	códigos	
LD HL , NN	33	a
LD BC , NN	1	a
LD DE , NN	17	a
LD IX , NN	221, 33	b
LD IY , NN	253, 33	b
LD SP , NN	49	— a

Tempo de execução destas instruções:

Grupo	N.º Bytes	N.º ciclos M	N.º ciclos T
a	3	3	10
b	4	4	14

Neste grupo, os 3 primeiros registros pares da tabela são denominados **utilitários** e os 3 últimos **especializados.** Por enquanto não vamos abórdar a utilização destes últimos.

Ensaio:

Programa 2 — carga de registros com dados memorizados nos endereços seguintes.

ZX81 e ((Spectrum))

mnemónicas

```
 10 CLS ((CLEAR 64999))
 20 LET A = 32412 ((CLS : LET A = 65000))
 30 POKE A, 14
 40 PRINT AT 5,0; "INTRODUZA UM VALOR (0 a 255)"
    "PARA O REGISTRO C"
 50 INPUT C
 60 POKE A + 1, C
 70 POKE A + 2, 6                          LD B, N
 80 POKE A + 3, 0
 90 POKE A + 4, 201                        RET
100 PRINT AT 12, 0; "O CONTEÚDO DO REGISTRO C"
    ,," = "; USR 32412 ((USR 65000))
111 RUN ((RUN 20))
```

Nota: Para o Spectrum, escrever ou alterar na linha respectiva, o que estiver entre parentesis **dobrados.**

Comentários: como já fizemos referência, o comando USR, antecedido duma instrução PRINT, inscreve em decimal o conteúdo do Registro BC, no regresso ao BASIC.

Depois deste ensaio, altere a linha 70 para POKE A + 2, 0 e corra o programa. Agora verifica-se um erro entre o valor introduzido no INPUT e o valor inscrito. Sabe explicar o que aconteceu?

Sim? ... Claro, é isso mesmo ...

Quando do salto para a rotina máquina, o registo BC recebe o endereço base dessa rotina. você apagou com o INPUT a parte "C" desse endereço, mas não apagou a parte "B" e ela regressa.

Programa 2 A — Idem

ZX81 e Spectrum

mnemónicas

```
 10 CLS ((CLEAR 64999))
 20 LET A = 32412 ((CLS : LET A = 65000))
 30 POKE A, 1                              LD BC, NN
 40 PRINT AT 5,0; "INTRODUZA UM VALOR (0 a
    65535)",,"PARA O REGISTRO BC"
 50 INPUT BC
 60 LET M = BC/256
 61 LET Y = INT M
 62 LET Z = ((M — Y) * 256)
 70 POKE A + 1, Z
 80 POKE A + 2, Y
 90 POKE A + 3, 201
100 PRINT AT 12,0; "O CONTEÚDO DO REGISTRO
    BC",," = "; USR 32412 ((USR 65000))
110 RUN ((RUN 20))
```

Comentários: 2 bytes (LOW DATA BYTE) e (HIGH DATA BYTE) encontrados no cálculo efectuado pelas linhas 60 a 62, vão carregar o Registro BC. De regresso ao Basic, esse número é inscrito em decimal.

Este grupo de instruções não afecta os flags.

(Cont. no próximo número)

HISTÓRIA DOS MICROS SINCLAIR

Trad. e Adapt. de "The Complete Sinclair Database"

PARTE I — 1962 - 1980

CLIVE MILES SINCLAIR (nascido em 1940), assistente editorial da revista «Practical Wireless», fundou em 1962 a firma «Sinclair Radionics» que comercializava pequenos transistores. Com a produção de novos equipamentos (rádio-transistores, micro-Tvs, calculadoras de bolso, relógios digitais, etc.), a companhia parecia expandir-se com muito êxito. Na realidade as coisas aconteceram ao contrário: deficiências nos equipamentos originaram o descrédito público na firma.

Sinclair mudou-se para a NEB (National Enterprise Board) criando aí o «design» de um computador chamado **NEWBRAIN,** que acabou por vender à firma «Newbury» por achá-lo muito dispendioso.

Em 1978, sinclair produzia efectivamente o seu primeiro computador — o **MK 14.**

Contendo um processador SC/MP CMOS, com 1/4 K RAM, um teclado hexadecimal e um monitor com 1/2 K EPROM, o MK 14 foi colocado à venda por £: 43.55. O seu concorrente era um COMMODORE que custava o dobro.

Algumas deficiências, uma das quais era o teclado ser fabricado em borracha, originaram grandes críticas na imprensa e o MK 14 perdeu a popularidade que tinha conquistado.

Em Fevereiro de 1980, Sinclair lançou o **ZX80,** o primeiro computador em BASIC, reproduzindo um processador da NEC — o Z80A.

Por um preço inferior a £ 100, o ZX80 vendia-se em forma de «Kit» ou já completo, incluindo um manual de programação.

Possuindo 1 K de memória RAM, o ZX80 era a primeira oportunidade de qualquer pessoa possuir e utilizar um computador.
O BASIC ocupava apenas 4 K ROM e as instruções introduziam-se nos programas por simples pressão de uma tecla.
As encomendas atingiram um número tal, que a Sinclair teve que determinar prazos de espera de 3 meses que, mesmo assim, nunca eram cumpridos.
O mais importante no ZX80 era que, não obstante possuir apenas 1 K de memória RAM, tinha bastantes aplicações sem exigir «extras».

Pouco tempo depois vendia-se uma extensão, de memória de 3 K RAM e, mais tarde, 16 K RAM. Mas rapidamente a máquina começou a demonstrar os seus limites, dado que o microprocessador apenas realizava uma tarefa de cada vez.
Aproveitando o «design» anterior, logo se idealizou uma cópia — **o MICRO ACE.**
Com circuitos principais e BASIC ROM idênticos ao ZX80, embora possuindo 2 K RAM (e não 1 K), esta máquina veio a ser vendida, sob licença, aos E.U.A.
Melhoramentos ao «design» foram prometidos mas, na realidade, nunca vieram a público...

(Cont. no próximo número)

CONVERSÃO DE PROGRAMAS DO ZX81→ZX SPECTRUM

Autor: FERNANDO PRECES

(Cont. dos números anteriores)

PARTE I

1.3 — PROGRAMA EM BASIC (Cont.)

Uma vez gravado o programa basic (v. número anterior), vamos introduzir na máquina o monitor que irá formar uma REM suportando 1751 bytes, necessários à introdução do código-máquina.

```
   1 REM ...
  10 REM X* STEP \:PNOT ABS (.!
MOVE \ STEP ■?? DIM :QNOT SGN >=
? FOR XXXXXXXXXXXXXXXXXXX
  20 LET a$="0422050920580801951
89040014035034205092062046033209
09220513601502423305808119518820
0024235"
  25 REM   Primeira REM com qual
quer extensao
  30 FOR f=1 TO LEN a$/3
  40 POKE 23769+f,VAL a$((3*f)-2
 TO 3*f)
  45 LET k=23769+f: PRINT k,PEEK
 k
  50 NEXT f
  51 STOP
  55 INPUT "QUANTOS BYTES ? ";a
  57 LET a=a+2
  60 LET b=INT (a/256): LET c=(a
/256-b)*256
  70 POKE 50000,c: POKE 50001,b
 600 RANDOMIZE USR 23770
1000 REM   DEPOIS DE OBTIDA A RE
SERVA DE BYTES NECESSARIA  AO CO
DIGO MAQUINA, DEVE APAGAR  TODAS
AS RESTANTES  LINHAS DO PROGRAMA
CONFORME INSTRUCOES DO TEXTO.
```

Após ter escrito o monitor, corra o programa com RUN. No ecran surge a pergunta: «Quantos Bytes?»
Como resposta, introduza o número 1751, aguarde o aparecimento do relatório e LIST o programa.
Apague todas as linhas do basic, excepto a linha REM.
Agora introduza:

POKE 23755, 39 ENTER
POKE 23756, 16 ENTER

Vamos chamar a seguir o programa basic «CONVERSOR» que se encontra gravado em fita, com:

MERGE «CONVERSOR»

Logo que surja o relatório, pode voltar a gravar o programa, agora já aumentado com a linha REM, com:

SAVE «CONVERSOR 1»

NOTA: VERIFIQUE o programa antes de o apagar da máquina, com:
VERIFY " "

Se tudo bateu certo, vamos continuar, com NEW e depois CLEAR 25984.
Seleccione a seu gosto um pequeno programa monitor que introduza código-máquina em decimal.
E agora vamos ao trabalho. Tem de escrever 1751 códigos, mas não o faça duma vez pois fica estoirado e vai cometer erros. Introduza 100 ou 200 códigos e depois pare. Pode gravar o que fez, apontando o último endereço e:

SAVE "0" code 25985,
(o número de códigos escritos)

a segunda etapa, com:

SAVE "1" code (número apontado+1),
(número de códigos escritos)

etc...

```
25985  39  16  46  0   253 168
25991  115 40  50  51  55  51
25997  50  14  0   0   180 92
26003  0   41  58  249 192 40
26009  168 83  40  50  51  54
26015  50  55  14  0   0   75
26021  92  0   41  45  57  56
26027  14  0   0   98  0   0
26033  41  13  39  17  93  0
26039  245 172 49  49  14  0
26045  0   11  0   0   44  57
26051  14  0   0   9   0   0
26057  59  220 49  14  0   0
26063  1   0   0   59  221 49
26069  14  0   0   1   0   0
26075  59  34  68  73  68  65
26081  84  73  67  32  83  79
26087  70  84  87  46  34  58
26093  245 58  245 34  32  67
26099  79  78  86  69  82  83
26105  79  82  32  32  90  88
26111  56  49  32  45  45  62
26117  32  90  88  83  80  69
26123  67  84  82  85  77  34
26129  58  234 13  39  18  53
26135  0   237 52  14  0   0
26141  4   0   0   58  249 192
```

```
26147  49   55   48   49   53   14
26153  0    0    0    91   0    58
26159  237  50   57   53   14   0
26165  0    39   1    0    58   250
26171  82   36   61   34   67   34
26177  203  236  54   48   48   14
26183  0    0    88   2    0    13
26189  39   19   13   0    236  49
26195  48   48   48   50   14   0
26201  0    18   39   0    13   243
26207  237  115  0    64   205  242
26213  67   203  250  0    0    205
26219  238  91   43   0    48   28
26225  62   0    187  48   19   35
26231  126  229  33   48   64   1
26237  0    1    237  177  225  62
26243  255  145  119  29   24   232
26249  203  254  203  186  205  52
26255  91   24   251  14   1    6
26261  0    62   127  219  254  31
26267  56   5    62   1    195  17
26273  66   0    23   23   23   210
26279  204  91   16   235  241  58
26285  25   64   254  0    40   5
26291  62   2    195  17   66   205
26297  240  91   205  52   91   203
26303  122  121  32   3    190  32
26309  199  35   23   48   241  17
26315  9    64   42   14   64   62
26321  1    50   25   64   205  52
26327  91   42   46   64   197  113
26333  19   6    80   75   62   0
26339  2    3    62   255  2    33
26345  45   64   205  37   66   33
26351  126  64   167  237  82   32
26357  23   213  62   3    17   38
26363  64   42   16   64   35   35
26369  35   237  160  237  160  61
26375  32   247  205  130  67   209
26381  193  48   8    42   42   64
26387  167  237  82   40   6    48
26393  167  62   4    24   9    121
26399  254  128  62   0    40   2
26405  62   5    195  17   66   213
26411  30   131  6    38   29   62
26417  255  219  254  23   23   203
26423  123  123  48   242  16   242
26429  209  63   32   4    254  76
26435  48   6    63   203  17   48
26441  1    201  195  54   91   42
26447  18   64   35   35   94   35
26453  35   62   0    187  201  0
26459  0    0    0    0    0    0
26465  0    0    0    0    0    0
26471  0    0    0    0    0    0
26477  0    0    0    0    0    0
26483  0    0    0    0    0    0
26489  0    0    0    0    0    0
26495  0    0    0    0    0    0
26501  0    0    0    0    0    0
26507  0    0    0    32   130  129
26513  131  136  138  137  139  143
26519  140  131  34   96   36   58
26525  63   40   41   62   60   61
26531  43   45   42   47   59   44
26537  46   48   49   50   51   52
26543  53   54   55   56   57   65
26549  66   67   68   69   70   71
26555  72   73   74   75   76   77
26561  78   79   80   81   82   83
26567  84   85   86   87   88   89
26573  90   165  166  167  0    0
26579  0    0    0    0    0    0
26585  0    0    0    0    0    0
26591  0    0    0    0    0    0
26597  0    0    0    0    0    0
26603  0    0    0    0    0    0
26609  0    0    0    0    0    0
26615  0    0    0    0    0    0
26621  0    11   10   8    9    0
26627  0    13   0    0    0    0
26633  0    0    0    14   0    143
26639  141  142  140  135  133  134
26645  139  143  131  140  34   96
26651  36   58   63   40   41   62
26657  60   61   43   45   42   47
26663  59   44   46   48   49   50
26669  51   52   53   54   55   56
26675  57   65   66   67   68   69
26681  70   71   72   73   74   75
26687  76   77   78   79   80   81
26693  82   83   84   85   86   87
26699  88   89   90   32   172  173
26705  0    175  176  177  178  179
26711  180  181  182  183  184  185
26717  186  187  188  189  190  192
26723  193  194  195  94   197  198
26729  199  200  201  203  204  205
26735  224  225  226  32   32   230
26741  83   232  233  234  235  236
26747  237  238  239  240  241  242
26753  243  244  245  80   247  248
26759  249  250  251  85   253  254
26765  255  42   28   64   34   30
26771  64   201  243  237  115  0
26777  64   0    0    0    0    0
26783  0    58   24   64   254  255
26789  32   24   58   26   64   254
26795  0    40   8    42   6    64
26801  205  190  65   24   7    237
26807  91   6    64   205  251  66
26813  24   3    205  143  67   42
26819  42   64   0    237  75   36
26825  64   55   237  66   48   5
26831  205  48   65   24   67   42
26837  40   64   167  237  66   32
26843  12   62   1    50   27   64
26849  42   28   64   34   8    64
26855  55   48   5    205  120  66
26861  24   211  42   38   64   55
26867  237  66   56   5    205  74
26873  68   24   241  0    205  224
26879  65   33   36   64   205  212
26885  65   24   186  205  236  65
26891  42   34   64   119  33   28
26897  64   195  48   68   0    0
26903  82   0    195  17   66   0
26909  84   93   1    0    0    167
26915  237  66   200  205  6    66
26921  205  170  65   27   98   107
26927  24   238  0    0    0    197
26933  78   35   70   3    112  43
26939  113  193  201  0    0    0
26945  0    33   32   64   205  37
26951  66   201  0    33   34   64
26957  205  37   66   229  213  42
26963  34   64   237  91   32   64
26969  55   237  82   209  225  192
26975  62   21   195  17   66   205
26981  224  65   42   32   64   126
26987  205  187  67   201  0    0
26993  0    0    50   24   64   237
26999  83   20   64   34   22   64
27005  237  123  0    64   251  201
27011  0    0    0    0    0    0
27017  0    213  229  94   35   86
27023  19   167  42   18   64   237
27029  82   32   6    237  91   10
27035  64   24   40   48   5    62
27041  13   195  17   66   42   14
27047  64   55   237  82   56   25
27053  0    42   12   64   237  82
27059  48   5    62   12   195  17
27065  66   42   10   64   237  82
27071  56   5    62   11   195  17
27077  66   225  115  35   114  43
27083  209  201  95   22   0    33
27089  48   64   25   126  201  0
27095  0    0    205  48   65   205
27101  6    66   254  128  200  79
27107  230  224  71   121  230  31
27113  246  32   205  111  66   50
27119  4    64   230  31   176  205
27125  170  65   230  224  254  96
27131  32   7    33   5    0    205
27137  190  65   201  254  160  32
27143  34   205  6    66   203  127
27149  32   10   205  111  66   246
27155  32   205  170  65   24   239
27161  230  127  205  111  66   246
27167  160  205  170  65   33   5
27173  0    205  190  65   201  254
27179  128  32   23   205  216  66
27185  235  205  190  65   201  205
27191  6    66   95   205  170  65
27197  205  6    66   87   205  170
27203  65   201  0    0    0    0
27209  254  224  32   37   33   17
```

```
27215  0    205  190  65   62   2
27221  205  170  65   201  0    0
27227  205  6    66   66   75   205
27233  111  66   80   89   205  170
27239  65   27   167  33   0    0
27245  237  82   32   234  201  0
27251  254  64   32   10   205  216
27257  66   205  167  67   205  251
27263  66   201  254  192  32   34
27269  205  216  66   27   213  205
27275  6    66   87   205  170  65
27281  30   2    205  6    66   205
27287  170  65   225  43   229  29
27293  32   244  21   32   239  209
27299  205  251  66   201  62   20
27305  195  17   66   237  115  0
27311  64   42   14   64   34   32
27317  64   237  75   30   64   237
27323  91   83   92   42   32   64
27329  126  18   205  224  65   19
27335  11   33   0    0    167  237
27341  66   32   238  42   8    64
27347  237  75   83   92   9    34
27353  75   92   195  36   68   0
27359  0    58   2    64   167  192
27365  42   40   64   0    34   42
27371  64   201  205  48   65   30
27377  4    1    4    64   197  213
27383  205  6    66   205  170  65
27389  209  193  2    3    29   32
27395  241  201  237  83   6    64
27401  56   3    64   254  0    200
27407  62   1    50   26   64   62
27413  255  195  17   66   33   36
27419  64   205  212  65   201  0
27425  0    0    0    0    0    0
27431  0    0    0    158  68   255
27437  79   1    150  1    150  128
27443  150  0    0    0    0    0
27449  0    0    0    0    0    0
27455  0    116  150  0    150  125
27461  64   0    0    0    0    0
27467  0    240  107  1    150  1
27473  45   0    17   3    64   33
27479  197  67   237  176  42   75
27485  92   1    183  0    9    34
27491  14   64   34   16   64   1
27497  249  106  9    43   34   18
27503  64   42   16   64   34   46
27509  64   43   34   34   64   1
27515  116  0    9    34   32   64
27521  201  235  54   128  35   34
27527  89   92   62   0    195  17
27533  66   205  212  65   42   89
27539  93   237  75   83   92   55
27545  237  66   68   77   42   28
27551  64   237  66   216  62   6
27557  195  17   66   62   0    50
27563  27   64   205  6    66   254
27569  118  32   26   62   13   205
27575  170  65   42   38   64   237
27581  75   36   64   167  237  66
27587  42   28   64   34   8    64
27593  200  205  143  67   201  254
27599  126  32   12   62   14   205
27605  170  65   33   5    0    205
27611  190  65   201  254  234  32
27617  15   62   234  205  170  65
27623  237  91   6    64   27   27
27629  205  167  67   201  0    0
27635  0    0    205  111  66   205
27641  170  65   201  42   75   92
27647  167  17   0    6    237  82
27653  1    0    1    17   0    91
27659  237  176  42   75   92   17
27665  0    5    167  237  82   1
27671  158  4    17   0    64   237
27677  176  0    42   75   92   17
27683  0    6    167  237  82   34
27689  75   92   54   128  35   34
27695  89   92   35   54   247  35
27701  54   13   35   54   128  35
27707  34   97   92   54   13   35
27713  54   128  201  0    0    0
27719  0    0    0    0    0    0
27725  0    0    0    0    0    0
27731  0    0    0    0    0    0
27737  0    0    0    0    0    0
```

(Cont. na próximo número)

OS PROCESSOS DA INTELIGÊNCIA ARTIFICIAL

O Professor António Costa, um brasileiro com profundos conhecimentos da Inteligência Artificial, fala dos principais acontecimentos nessa área, ocorridos nos últimos dois anos:

1. O Japão lança um projecto com 10 anos de duração, visando a construção de um computador da 5.ª geração. Ele será capaz de falar inglês e japonês, fazer traduções, programar outros computadores, observar cenas, etc.;

2. Os Estados Unidos entram na corrida para construir um computador de 5.ª geração, reservando verbas fabulosas para pesquisas em Inteligência Artificial só o Pentágono detém 1 bilhão de dólares para gastar em cinco anos. Não deixa de ser lamentável que parte das pesquisas em Inteligência Artificial seja financiada por instituições militares;

3. Bancos de Dados capazes de entender Inglês, Japonês, Alemão, Francês e até Português começam a ser vendidos nos Estados Unidos. Entre os utilizadores de tais Bancos estão o Bank o America e a AVCO;

4. A Xerox e a Symbolics Inc. construiram computadores especialmente projectados para executar programas em LISP, a principal Linguagem da Inteligência Artificial;

5. Os japoneses anunciam para 1985 um computador pessoal capaz de executar eficazmente programas em PROLOG, a linguagem que ocupa o segundo lugar em popularidade entre as pessoas que trabalham com Inteligência Artificial.

MASTER MIND

ZX81

Autor: FERNANDO PRECES

```
   1>REM "F1J1"
   5 GO SUB 450
  10 CLS
  15 DIM C(4)
  20 DIM G(4)
  30 LET C(1)=INT (RND*9)+1
  40 FOR Z=2 TO 4
  50 LET C(Z)=INT (RND*9)+1
  60 FOR J=1 TO Z-1
  70 IF C(J)=C(Z) THEN GO TO 40
  80 NEXT J
  90 NEXT Z
 100 FOR G=1 TO 10
 105 IF G<10 THEN PRINT "█TENTAT
IVA█";CHR$ (G+156);">"
 107 IF G=10 THEN PRINT "ULTIMA█
TENTATIVA>"
 110 INPUT A
 120 LET A1=A
 130 FOR Z=1 TO 4
 140 LET G(Z)=A-10*(INT (A/10))
 150 LET A=INT (A/10)
 160 NEXT Z
 170 LET B=0
 180 FOR Z=1 TO 4
 190 LET W=0
 200 IF C(Z)<>G(Z) THEN GO TO 23
0
 210 LET B=B+1
 220 LET G(Z)=0
 230 NEXT Z
 240 FOR Z=1 TO 4
 250 IF G(Z)=0 THEN GO TO 300
 260 FOR J=1 TO 4
 270 IF C(Z)<>G(J) THEN GO TO 29
0
 280 LET W=W+1
 290 NEXT J
 300 NEXT Z
 310 PRINT A1;" ACERTOU ";CHR$ (
B+156);" PRETA";
 320 IF B=1 THEN PRINT " ";
 330 IF B<>1 THEN PRINT "S";
 340 PRINT "  ";CHR$ (W+156);" B
RANCA";
 350 IF W<>1 THEN PRINT "S";
 360 PRINT
 370 IF B=4 THEN PRINT TAB (5);"
█VOCE█ACERTOU█";
 380 IF B=4 THEN GO TO 410
 390 NEXT G
 400 PRINT TAB 3;"A█RESPOSTA█COR
RECTA█ERA█";
 410 FOR Z=1 TO 4
 420 PRINT CHR$ (C(5-Z)+156);
 430 NEXT Z
 440 GO TO 600
 450 PRINT AT 0,8;"██MASTERMIND█
█"
 460 PRINT AT 2,0;" FUI PROGRAMA
DO PARA ESCOLHER AOACASO UM NUME
RO DE 4 DIGITOS."
 470 PRINT ,," SAO VALIDOS TODOS
 OS ALGARISMOS(0/9). NAO HA ALGA
RISMOS  REPETIDOS NO NUMERO ESCO
LHIDO."
 475 PRINT ,," VOCE VAI TENTAR
ADIVINHAR ESSENUMERO  INTRODUZIN
DO 4 DIGITOS ASUA ESCOLHA  POR C
ADA TENTATIVA,AO FIM DA QUAL AGU
ARDA O RESULTADO."
 480 PRINT ,," A INFORMACAO   PR
ETA  SIGNIFICA"
 490 PRINT "QUE TEM UM DIGITO  C
ERTO NA CASACERTA E  BRANCA , UM
 CERTO EM CASA ERRADA."
 550 PRINT ,,"  CARREGUE  N/L  P
ARA COMECAR:"
 570 INPUT U$
 580 CLS
 590 RETURN
 600 PRINT AT 21,0;" OUTRO JOGO?
 (DIGA S OU N)."
 610 INPUT L$
 620 IF L$="S" THEN GO TO 10
 630 CLS
 640 PRINT AT 12,3;" ENTAO ATE A
 PROXIMA."
 650 STOP
1000 SAVE "1"
1010 RUN
```

DEC-HEX

TS 1000/ZX81/SPECTRUM

Autor: CARLOS MORENO
Porto

"Este programa, tal como apresenta a listagem, funciona apenas no ZX81 ou no TS 1000. Apesar disso, podemos pô-lo a funcionar no ZX SPECTRUM com as seguintes alterações:

Linha 12 será: 12 LET N = 23760
Linha 61 será: 61 LET N = 23760

Outro modo possível de alterar este programa para o SPECTRUM é:

Linha 10 será: 10 CLEAR 5000
Linha 12 será: 12 LET N = 50001
Linha 61 será: 61 LET N = 50001

Ao ser executado, o programa passa os números introduzidos de hexadecimal para decimal, colocando-os em seguida na instrução REM. Para o microprocessador poder executar esta tarefa, devemos fazer RAND USR 16514. Atenção!... se for introduzido um Código Hexadecimal maior que FF, será produzida uma mensagem de erro e o programa pára a sua execução.

```
  10 REM .........................
................................
..........................
  12 LET N=16514
  15 LET L1=1
  25 PRINT "CODIGO HEX EM ";N+L1
-1;" ?"
  30 INPUT B$
  31 LET P=1
  32 LET R=0
  33 LET L=LEN B$
  34 FOR K=L TO 1 STEP -1
  35 LET J=0
  36 IF CODE B$(K)>37 THEN LET I
=7
  37 LET A=(CODE (B$(K))-28-J)*P
  38 LET P=P*16
  39 LET R=R+A
  40 NEXT K
  47 CLS
  48 IF B$="" THEN GOTO 62
  49 POKE N+L1-1,R
  50 LET L1=L1+1
  60 GOTO 20
  61 LET N=16514
  62 PRINT "PREPARE O GRAVADOR"
  63 PAUSE 4E4
  64 CLS
  65 SAVE "CARREGADOR".
  70 FOR F=N TO N+L1-1
  80 PRINT F,PEEK F
  90 NEXT F
```

EVOLUÇÃO

SPECTRUM 16/48 K

In. THE BEST OF SINCLAIR PROGRAMS/84
Trad.: J. MAGALHÃES

Aqui está um programa para quem gostar de Biologia, mais concretamente sobre evolução, selecção natural.
Em determinada população de ratos distinguem-se uns de côr preta, outros amarelos e, por «selecção», alguns serão eliminados enquanto outros poderão sobreviver.
A selecção não segue qualquer fórmula pois iria afastar-se bastante da realidade. Assim, será determinada ao acaso pela instrução RANDOM, mas dependente da percentagem que pretender.

```
   1 DIM a$(250): DIM b(51)
  10 GO SUB 9000
  15 BORDER 7: PAPER 7: CLS
  20 PRINT INK 1;"Este programa
simula determinadapopulacao   de
  ratos,  onde  seDistinguem doi
s tipos:"'"ratos pretos ▲ "'"ra
tos amarelos"; INK 6;"▲"'' INK 2
;"Esta  diferenca e  originada p
orum gene com dois alelos."'"Y (
preto) e"'"dominante sobre y (am
arelo)."
  23 PRINT '"Isto significa que:
"'"YY -- Preto"'"Yy -- Preto"'"y
y -- Amarelo"''; INK 1;"Nota que
 um ▲ rato pode ser YY  ou Yy de
 cor amarela"'"Os alelos amarelo
s podem ocultar-se em ratos pret
os!": PRINT #0; INK 0; INVERSE 1
;"Qualquer tecla para continuar"
  25 IF INKEY$="" THEN GO TO 25
  27 CLS : PRINT INK 2;"Podes ac
ompanhar as mutacoes     desta po
pulacao ate 50 geracoes."'' INK
1;"Em cada geracao a populacao
   dobra o seu numero, mas nem
   todos podem sobreviver. Seras
   interrogado sobre qual o nume
ro maximo possivel de sobreviven
tes"
  28 PRINT INK 2;'"Sao possiveis
 3 situacoes:"'"- Seleccao nao f
avoravel aos     amarelos"'"(Os p
retos tem  a cor que lhes  torna
 possivel a sobrevivencia)"'"- S
eleccao nao favoravel aos     rat
os pretos"'"(mais facil a sobrev
ivencia dos ratos amarelos"'"- N
ao existe seleccao"'"(Os dois ti
pos de ratos adaptam--se perfeit
amente ao meio)"
  29 PRINT #0; INVERSE 1;"Qualqu
er tecla para continuar": PAUSE
0
  30 CLS : PRINT INK 2;"Podes te
ntar qualquer uma destassituacoe
s, e se ocorrer seleccaodetermin
ar o seu numero"' INK 0; INVERSE
 1;'"Qualquer tecla para comecar
."
  35 IF INKEY$="" THEN GO TO 35
  40 BORDER 7: PAPER 7: CLS
  50 PRINT AT 7,0; INK 2;"Popula
cao maxima?"'"(Nao superior a 12
5)"
  52 INPUT p0: IF p0<1 OR p0>125
 THEN PRINT INK 0;''"Deve ser en
tre 0 a 125.": GO TO 52
  55 CLS : PRINT AT 7,0; INK 1;"
A seleccao pode ser:"''"Nao favo
ravel aos amarelos- ""0"""'"Nao
favoravel aos pretos-   ""1"""'"
Nao ha seleccao-             ""2"
""
  57 INPUT d: IF d<>0 AND d<>1 A
ND d<>2 THEN PRINT INK 0;''"Deve
 ser 0,1 ou 2": GO TO 57
  58 IF d=2 THEN GO TO 75
  60 CLS : PRINT AT 7,0; INK 2;"
Populacao seleccionada ?"'"(em p
ercentagem)"'"100% significa que
 toda a        populacao nao sobr
eviveria."'"De entrada de um nu
mero ate 100"
  62 INPUT sp: IF sp<0 OR sp>100
 THEN PRINT INK 0''"Deve ser ent
re 0 e 100": GO TO 62
  70 LET s=100-sp
  75 CLS : PRINT AT 7,0; INK 1;"
Qual a percentagem inicial
de alelos amarelos ?  (y%)"''"(N
ota que muitos deles podem     so
frer alteracoes na cor!)"''"Um n
umero ate 100"
  77 INPUT y: IF y<0 OR y>100 TH
EN PRINT INK 0''"Deve ser entre
0 e 100": GO TO 77
 100 LET n=1: LET p=p0
 110 LET y=y/100: LET b(1)=y
 200 LET sb=1: LET sy=1: IF d=0
THEN LET sy=s/100
 201 IF d=1 THEN LET sb=s/100
 215 BORDER 7: PAPER 7: CLS
 220 LET n=n+1: GO SUB 1000
 265 GO SUB 2000
 268 IF n=51 THEN PAUSE 120: GO
TO 310
 270 POKE 23692,255
 280 PRINT #0; BRIGHT 1; INVERSE
 1; INK 0;"M - CONTINUAR       G
 - GRAFICO"
 285 GO SUB 5000
 290 IF I$="m" OR i$="M" THEN GO
 TO 215
 300 IF i$="g" OR i$="G" THEN GO
 TO 310
 305 GO TO 285
 310 GO SUB 6000
 311 GO SUB 5000
 315 IF i$="m" OR i$="M" THEN GO
 TO 215
 320 IF i$="n" OR i$="N" THEN GO
 TO 40
 325 IF i$="s" OR i$="S" THEN PA
PER 7: STOP
 350 GO TO 311
1000 PRINT AT 0,7; INK 2; BRIGHT
 1;"PROXIMA GERACAO (";N-1;")"
1005 LET ym=0: LET cd=0: FOR f=1
 TO 2*p
1010 LET x=RND: LET v=RND
1020 LET z=(x>=y)+(v>=y)
1030 LET cd=cd+z
1040 LET a$(f)=STR$ (z)
1045 LET i=0: IF a$(f)="0" THEN
LET i=6: LET ym=ym+1
1046 PRINT INK i;"▲ ";
1050 NEXT f
1060 LET y=(4*p-cd)/(4*p)
1065 PRINT AT 19,0; INK 2; BRIGH
T 1;ym;" ratos amarelos em  ";(
2*p)'(4*p-cd);" alelos amarelos
em ";(4*p)
1070 RETURN
2000 PAUSE 150: PRINT AT 0,2; IN
K 2; BRIGHT 1; FLASH 1;" NEM TOD
OS PODEM SOBREVIVER "
2005 LET ps=2*p: LET bm=2*p-ym:
LET bm1=bm: LET ym1=ym
2006 LET r=p0/(sb*bm+sy*ym)
2007 IF r>1 THEN LET r=1
2010 FOR f=1 TO 2*p
2060 LET x=RND
2071 IF ((a$(f)>"0") AND (x>r*sb
)) THEN LET cd=cd-VAL (a$(f)): L
ET bm1=bm1-1: LET a$(f)="d": LET
 ps=ps-1: PRINT "  ";
```

```
2072 IF ((a$(f)="0") AND (x>r*sy
)) THEN LET ym1=ym1-1: LET a$(f)
="d": LET ps=ps-1: PRINT " ";
2075 IF a$(f)<>"d" THEN PRINT OV
ER 1; INK 8;" ";
2080 NEXT f
2082 PRINT AT 20,31;" "
2083 PRINT AT 0,0;"
"
2084 IF bm=0 THEN PRINT AT 19,0;
"
2085 IF bm>0 THEN PRINT AT 19,0;
 BRIGHT 1; INK 2;bm1;" PRETOS SO
BREV. DOS ";bm; INK 1;TAB (28);I
NT (100*bm1/bm+.5);"%"
2086 IF ym>0 THEN PRINT AT 20,0;
 BRIGHT 1; INK 2;ym1;" AMARELOS
SOBREV. DOS ";ym; INK 1;TAB (28)
;INT (100*ym1/ym+.5);"%"
2090 LET y=(2*ps-cd)/(2*ps): PRI
NT BRIGHT 1; INK 1;"%y=";(INT (y
*1000+.5))/10;"   (ultima geraca
o =";(INT (b(n-1)*1000+.5))/10;"
)"
2095 LET b(n)=y
2098 LET p=p0: IF ps<=p0 THEN LE
T p=ps
2100 RETURN
5000 IF INKEY$="" THEN GO TO 500
0
5020 LET i$=INKEY$: RETURN
6000 BORDER 1: PAPER 1: CLS : IN
K 7: PLOT 47,156: DRAW 0,-120: D
RAW 200,0
6010 FOR f=36 TO 156 STEP 24: PL
OT 45,f: DRAW -3,0: NEXT f
6020 FOR f=47 TO 247 STEP 40: PL
OT f,36: DRAW 0,-4: NEXT f
6030 PRINT AT 9,0;"%y";AT 2,2;"1
00";AT 5,3;"80";AT 8,3;"60";AT 1
1,3;"40";AT 14,3;"20";AT 17,4;"0
"
6040 PRINT AT 18,6;"0   10   20
  30   40   50";AT 19,15;"Geraca
o"
6050 PRINT AT 1,8; INVERSE 1;"Po
pulacao max. ";p0
6060 IF d=2 THEN PRINT AT 0,10;
INVERSE 1;"Nao houve seleccao  "
6070 IF d=1 THEN PRINT AT 0,2; I
NVERSE 1;INT (sp);"% nao a favor
 dos pretos"
6080 IF d=0 THEN PRINT AT 0,1; I
NVERSE 1;INT (sp);"% nao a favor
 dos amarelos"
6100 PLOT 46,120*b(1)+36: DRAW O
VER 1;3,0
6110 FOR f=2 TO n: PLOT OVER 1;4
2+4*f,120*b(f)+36: NEXT f
6115 IF n=51 THEN GO TO 6200
6120 PRINT AT 21,0; INVERSE 1;"T
ecla ""M"" CONTINUAR": PRINT #0;
 INVERSE 1;"""N"" - COMECAR
""S"" - PARAR"
6130 INK 0: RETURN
6200 LET n=1: LET b(1)=y
6210 PRINT AT 19,0; INVERSE 1; F
LASH 1;"Grafico completo"' FLASH
 0;"M- continua mas limpa o graf
ico"'"(n- comecar de novo, s- pa
rar)"
6215 INK 0: RETURN
9000 DATA 0,BIN 00001001,BIN 000
00110,BIN 00111110,BIN 01111100,
BIN 01111100,BIN 10000000,BIN 11
111110
9010 FOR f=0 TO 7
9015 READ gr
9020 POKE USR "m"+f,gr
9025 NEXT f: RETURN
```

VU-METER II

SPECTRUM 16 K

In. YOUR COMPUTER, Dezembro/83
Trad.: J. MAGALHÃES

VU-METER II

No n.º de Dezembro/83 do CLUBE Z80, publicamos o programa VU-METER (corrigido) que nos dava a representação gráfica de sinais audio.
Devido ao entusiasmo demonstrado por vários sócios, decidimos publicar este novo programa que funciona de forma idêntica ao anterior, mas com pequenas alterações.
Temos agora 3 tipos de representações gráficas:

— Normal
— Barras
— Pontos

que supomos tornarem mais fácil a observação das diferentes tonalidades.

```
 102 DEF FN A(L)=1+INT (.5+L/30)
 103 GO SUB 405
 105 GO TO 200
 110 DRAW INK FN A(L);X-N,-L+Y:
LET X=N: LET Y=L: RETURN
 120 DRAW INK FN A(L);0,-L: RETU
RN
 200 PAPER 0: INK 7: BRIGHT 1: C
LS
 210 PRINT AT 1,9; INVERSE 1;"SC
ANNER DE SOM"
 215 PRINT AT 3,1;"Este programa
 faz a representacao grafica de
sinais audio pela entrada do EAR
 do Spectrum"
 220 PRINT AT 8,1;"Introduza uma
 cassete (ex: musica), inicie o
gravador e seleccione uma das se
guintes opcoes:"
 225 PRINT AT 13,8;"1) NORMAL";A
T 15,8;"2) BARRAS";AT 17,8;"3) P
ONTOS"
 230 INPUT INVERSE 1;"Qual a opc
ao (1-3)";Q: IF Q<1 OR Q>3 THEN
GO TO 230
 250 CLS : PRINT AT 0,7; INVERSE
 1;"SPACE para parar": LET I$=""
: LET X=0: LET Y=0
 255 FOR N=0 TO 255 STEP 4: LET
L=USR TONE: PLOT N,L
 260 IF Q<3 THEN GO SUB 100+(Q*1
0)
 265 LET I$=INKEY$: IF I$=" " TH
EN GO TO 300
 270 NEXT N
 275 GO TO 250
 300 PRINT AT 0,5; INVERSE 1;"M=
MENU  R=RECOMECAR  E=FIM": PAUSE
 0
 305 LET I$=INKEY$: IF I$="M" TH
EN RUN
 310 IF I$="R" THEN GO TO 270
 315 IF I$="E" THEN STOP
 320 GO TO 300
 400 DATA 1,0,255,17,0,0,219,254
,203,119,32,1,19,16,247,66,75,20
1
 405 LET TONE=32600
 410 FOR N=TONE TO TONE+17: READ
 D: POKE N,D: NEXT N: RETURN
```

PROCESSADOR DE TEXTO

SPECTRUM 16/48 K

In. YOUR COMPUTER, Março 1983
Adapt.: J. MAGALHÃES

* Este texto foi elaborado com o programa que a seguir passamos a descrever.
Retirado da revista Your Computer, Marco de 1983.
Por falta de disponibilidade, o programa e apresentado com todas as instrucoes em Ingles, no entanto supomos que a nossa descricao e bastante elucidativa, facilitando a sua utilizacao.

Como ja teve oportunidade de verificar, este "processador de texto", permite ultrapassar o limite normal de caracteres do Spectrum (32 - 42), dando-lhe ainda a possibilidade de uma nova redefinicao. Neste caso estao a ser usados caracteres de seis por oito pixels.

O programa inicialmente apresenta um menu de seis opcoes:
- a primeira opcao, da entrada para a criacao do texto, ou verificacao do mesmo.
Inicialmente e-lhe pedido o n. da pagina pretendida que sera apresentada seguidamente com o cursor posicionado no canto superior esquerdo do ecran, podendo, com as teclas 5 a 8, move-lo para qualquer posicao. Se acontecer sobreposicao de caracteres, ficarao na memoria apenas os ultimos a ser transcritos.
Sao possiveis todos os caracteres, para o qual pode utilizar as teclas: Caps shift e symbol shift. Em caso de erro na entrada de um caracter, para o apagar basta usar, como normalmente as teclas Caps shift e Delete.
Se pretender iniciar uma nova linha, faca Enter, no entanto note que se estiver na ultima linha, nao avancara para a proxima pagina, ficando o cursor dividido em duas partes.
Verifique que no fundo do ecran se encontram ainda mais tres opcoes:
Next- Passara a pag. seguinte. (sao apenas utilizadas 4 pags).
Menu- Para regressar ao Menu.
Copy- Fazer a copia do texto.(tecla Z).
Estas opcoes sao conseguidas accionando simultaneamente as teclas: Caps Shift e Symbol shift, seguindo-se a entrada da sua inicial (inverse video).

E apresentado um outro menu ao fazer Edit (Caps shift + tecla 1), do qual constam as seguintes opcoes:
Insert- Para introducao de caracteres nao introduzidos por engano ou erro. (Tecla I)
Posicione o cursor no local indicado (para correccao) e pelo modo Edit, accione a tecla "I". Introduza o numero de caracteres a entrar, num limite de 0 a 255. Pode fazer esta entrada de duas formas: ex. 1 ou 001. Neste ultimo exemplo a instrucao Enter e feita automaticamente.
Delete- esta opcao e identica a anterior sendo utilizada para retirar caracteres na linha, onde ocorreu o erro. E pretendido tambem o numero de caracteres a sair, sendo esta operacao feita pelo mesmo modo da opcao anterior.
Erase- Limpa determinada parte do texto, desde a ultimo caracter a linha imediatamente abaixo do posicionamento do cursor, quando requerida esta opcao.
Justify l/r- Funciona para acerto de uma linha ou de toda a pag. (cursor line ou page, respectivamente). Experimente num pequeno texto deixar dois espacos antes de iniciar uma linha. Accione a tecla "l" e indique se pretende a correcao nessa linha ou em todo o texto (c/p). Coloque, antes desta opcao o cursor no inicio do texto, se optou por; page (p), ou na linha se optou por linha (c). Verifique que o texto ou linha, ficaram juntas a margem esquerda Pela opcao "r" (direita), conseguira o acerto de linhas entre as duas margens do ecran, pela mesma forma da opcao anterior. (Todo o texto ou uma linha).

Voltando ao Menu principal tem ainda as opcoes:
Stop- Parar o programa. (Com a instrucao "Continue" regressara ao menu). Em caso de brake ao programa, faca Goto 9000 e nao Run.
Erase a page- Apagara toda a pag. indicada (1-4)
Save e Load- Funcionam da mesma forma de gravacao e carregamento de um programa normal, atribuindo um nome que nao deve exceder os 10 caracteres.
Change typeface- Contem tres opcoes: Normal, New e Redefine.
Normal, tera os caracteres normais do Spec

(Maiusculas).
New, para utilizar os caracteres previamente definidos, pela opcao Redefine.
Redefine- Para redefinir qualquer, de entrada do caracter, seguindo-se o codigo decimal correspondente. (Ver tabela de Conversao - Jornal n. 20 Clube Z80).
Assim que o caracter vai sendo definido, podera acompanhar a sua formacao pelo display apresentado numa "grelha" de 1-8 linhas com colunas de a-f. Se pretender ficar com o quadro dos novos caracteres deve gravar o programa com o comando GOTO 9900.

Como carregar o programa ?

Carregue o programa em BASIC (list. 1) e grave-o com o comando: GOTO 9900; no final da gravacao o programa fara "auto-run", ficando no modo LOAD; faca BREAK e verique a gravacao (VERIFY "").
Para introducao do codigo maquina, utilize um "carregador" a sua escolha, nao esquecendo fazer CLEAR 29665, antes de iniciar esta operacao.
Utilize por exemplo, o carregador do c.m. publicado no boletim n. 12 do CLUBE Z80, pag. 14 "FIRE FOX").
Para gravacao do cod. maq. (depois de introduzido completamente), faca:
SAVE "spc"CODE 29666,3102; verifique.
Note: O programa Basic deve estar gravado na cassete antes do cod. maq.
Depois de completas todas as operacoes, com o cod. maq. intruduzido sem erros, com o programa gravado devidamente, pode verificar o seu trabalho.
Inicie o gravador com a cassete no principio do programa e faca: LOAD "", o cod. maq. entrara automaticamente.
O programa comeca pela apresentacao do Menu principal. BOM TRABALHO !

Listagem 1

```
  10 POKE 23606,1: RANDOMIZE USR
 31013: DIM x$(4,924): LET x=0:
LET t=50: LET v=23560: LET m=900
0: GO TO m
  50 RANDOMIZE USR 31083: INPUT
INKEY$: PRINT #x;"Page(";PEEK 31
619;") █ext : █enu : █copy"
  55 RANDOMIZE USR 30821
  60 RANDOMIZE USR 29951
 140 IF PEEK v=7 AND PEEK 32651<
>43 THEN GO TO 700
 150 IF PEEK v<>14 THEN GO TO 60
 180 RANDOMIZE USR 29971
 230 IF PEEK v=CODE "n" THEN RAN
DOMIZE USR 31038: GO TO t
 250 IF PEEK v=CODE "m" THEN GO
TO m
 270 GO TO 60
 540 RANDOMIZE USR 31083: INPUT
INKEY$: PRINT #x;"How many Chrs.
 ? ";
 570 RANDOMIZE USR 29666: RANDOM
IZE USR 30821: IF PEEK 29854=x T
HEN GO TO 780
 600 RANDOMIZE USR 31500: GO TO
t
 700 RANDOMIZE USR 32592
 710 RANDOMIZE USR 31083: INPUT
INKEY$: PRINT #x;"█nsert:█elete:
█rase:JUSTIFY,█/█"
 730 RANDOMIZE USR 30821: PAUSE
x: IF PEEK V=CODE "i" THEN POKE
31544,10: GO TO 540
 740 IF PEEK v=CODE "d" THEN POK
E 31544,x: GO TO 540
 750 IF PEEK v=CODE "e" THEN RAN
DOMIZE USR 31242: GO TO 780
 760 IF PEEK v=CODE "l" THEN POK
E 31412,79: GO TO 8000
 770 IF PEEK v=CODE "r" THEN POK
E 31412,x: GO TO 8000
 780 RANDOMIZE USR 32592: GO TO
t
1000 RANDOMIZE USR 29900: GO TO
t
2000 RANDOMIZE USR 30434: STOP
2010 GO TO m
3000 RANDOMIZE USR 29915
3020 IF PEEK v>CODE "4" THEN GO
TO m
3050 RANDOMIZE USR 30545: GO TO
5020
4000 RANDOMIZE USR 30615: INPUT
i$
4010 PRINT AT 11,x;"SAVING:""";i
$;""" DATA x$()": SAVE i$ DATA x
$(): GO TO 5020
5000 RANDOMIZE USR 30674: INPUT
i$
5010 RANDOMIZE USR 30720: PAUSE
x: LOAD i$ DATA x$()
5020 RANDOMIZE USR 30756: PAUSE
x: GO TO m
6000 RANDOMIZE USR 29927
6030 IF PEEK v=CODE "1" THEN RAN
DOMIZE USR 32573: GO TO m
6050 IF PEEK v=CODE "2" THEN RAN
DOMIZE USR 32580: GO TO m
6055 RANDOMIZE USR 30200
6060 INPUT i$
6080 IF LEN i$>1 OR i$<" " OR i$
>"©" THEN RANDOMIZE USR 31684: G
O TO 6060
6085 POKE 30177,CODE i$
6090 RANDOMIZE USR 30076
6100 FOR b=1 TO 8
6110 RANDOMIZE USR 30157
6140 RANDOMIZE USR 29666: RANDOM
IZE USR 29978: NEXT b
6190 PRINT #x;"OK? █epeat :Oth
er keys MENU": RANDOMIZE USR 308
21: PAUSE x: IF INKEY$="r" THEN
CLS : GO TO 6055
6200 GO TO m
8000 RANDOMIZE USR 31083: INPUT
INKEY$: PRINT #x;"█age or █ursor
 line ?"
8020 RANDOMIZE USR 30821: PAUSE
x: IF PEEK v=CODE "c" THEN RANDO
MIZE USR 31214: GO TO t
8030 IF PEEK v=CODE "p" THEN RAN
DOMIZE USR 31225: GO TO t
8040 GO TO 780
9000 RANDOMIZE USR 29939: GO TO
1e3*(PEEK v-48)
9800 CLEAR 29665: LOAD ""CODE 29
666: GO TO 10
9900 SAVE "type" LINE 9800: SAVE
 ""CODE 29666,3102
```

Listagem 2

```
29666   42  134  92   34   159  116
29672   42  138  92   34   161  116
29678   33  155  116  6    1    229
29684   197 205  163  116  193  225
29690   58  8    92   254  13   40
29696   29  119  214  48   56   126
29702   214 10   48   122  197  229
29708   62  253  205  1    22   225
29714   229 126  215  225  193  35
29720   4   120  254  4    32   213
29726   5   40   99   43   126  214
29732   48  50   158  116  5    40
29738   54  43   126  214  48   135
29744   79  135  135  129  79   58
29750   158 116  129  50   158  116
29756   5   40   34   43   126  214
29762   48  40   28   254  1    32
29768   4   14   100  24   13   254
29774   2   32   51   58   158  116
29780   214 56   48   44   14   200
29786   58  158  116  129  50   158
29792   116 58   158  116  50   41
29798   123 50   226  117  205  176
29804   123 6    32   118  16   253
29810   6   2    205  68   14   42
29816   159 116  34   134  92   42
29822   161 116  34   138  92   201
29828   205 196  123  205  119  116
```

```
29834 62 253 205 1 22 175
29840 215 175 215 175 215 205
29846 119 116 195 226 115 49
29852 48 49 1 242 80 15
29858 23 118 253 203 1 110
29864 40 249 253 203 1 174
29870 201 205 196 123 205 163
29876 116 58 8 92 214 49
29882 56 243 33 203 116 150
29888 48 237 6 24 205 68
29894 14 205 176 123 201 4
29900 205 174 118 62 4 50
29906 203 116 205 178 116 205
29912 225 126 201 205 43 119
29918 62 255 50 203 116 205
29924 178 116 201 205 111 120
29930 62 3 50 203 116 205
29936 178 116 201 205 10 121
29942 62 6 50 203 116 205
29948 178 116 201 205 80 127
29954 205 163 116 205 121 121
29960 56 8 92 254 12 192
29966 205 234 126 24 239 205
29972 163 116 205 201 121 201
29978 205 68 127 42 228 117
29984 58 226 117 119 35 34
29990 228 117 62 2 205 1
29996 22 62 22 215 62 15
30002 215 62 19 215 62 21
30008 215 175 215 58 225 117
30014 215 58 183 117 60 50
30020 183 117 58 226 117 135
30026 50 226 117 48 22 62
30032 2 205 1 22 62 22
30038 215 58 183 117 215 58
30044 227 117 215 62 143 215
30050 205 234 126 58 227 117
30056 60 254 19 40 8 50
30062 227 117 58 226 117 24
30068 212 205 113 121 205 176
30074 123 201 175 50 183 117
30080 17 224 122 38 0 58
30086 225 117 111 41 41 41
30092 25 34 228 117 62 2
30098 205 1 22 62 22 215
30104 62 15 215 62 15 215
30110 58 225 117 215 205 68
30116 127 62 22 215 62 15
30122 215 62 19 215 58 225
30128 117 215 205 113 121 201
30134 22 8 0 32 32 32
30140 32 32 32 32 32 32
30146 13 32 68 69 67 46
30152 32 78 111 46 62 62
30158 2 205 1 22 17 182
30164 117 1 23 0 205 60
30170 32 62 11 50 227 117
30176 201 49 1 18 112 124
30182 16 0 22 0 10 48
30188 16 6 143 143 143 143
30194 143 143 16 5 143 143
30200 205 107 121 62 2 205
30206 1 22 6 8 197 58
30212 233 117 60 50 233 117
30218 58 235 117 60 50 235
30224 117 17 230 117 1 18
30230 0 205 60 32 193 16
30236 229 175 50 233 117 62
30242 48 50 235 117 24 101
30248 16 0 22 11 11 78
30254 69 87 32 120 56 22
30260 9 11 97 98 99 100
30266 101 102 22 15 1 83
30272 80 69 67 84 82 85
30278 77 32 67 72 82 46
30284 32 63 32 61 32 32
30290 32 39 78 69 87 39
30296 32 77 79 68 69 22
30302 1 20 97 61 49 50
30308 56 22 2 20 98 61
30314 32 54 52 22 3 20
30320 99 61 32 51 50 22
30326 4 20 100 61 32 49
30332 54 22 5 20 101 61
30338 32 32 56 22 6 20
30344 102 61 32 32 52 62
30350 2 205 1 22 17 40
30356 118 1 101 0 205 60
30362 32 201 22 11 8 80
30368 65 71 69 32 82 69
30374 81 85 73 82 69 68
30380 32 63 62 2 205 1
30386 22 17 156 118 1 18
30392 0 205 60 32 205 101
30398 120 201 22 11 0 67
30404 79 78 84 73 78 85
30410 69 32 58 32 119 105
30416 108 108 32 114 101 115
30422 116 97 114 116 32 97
30428 116 32 77 69 78 85
30434 62 2 205 1 22 17
30440 192 118 1 34 0 205
30446 60 32 205 101 120 201
30452 22 11 2 80 65 71
30458 69 32 78 111 46 32
30464 116 111 32 98 101 32
30470 69 82 65 83 69 68
30476 32 49 45 52 32 63
30482 22 13 5 65 78 89
30488 32 79 84 72 69 82
30494 32 75 69 89 32 70
30500 79 82 32 77 69 78
30506 85 62 2 205 1 22
30512 17 244 118 1 55 0
30518 205 60 32 205 101 120
30524 201 22 15 7 80 65
30530 71 69 32 78 79 46
30536 32 32 32 69 82 65
30542 83 69 68 205 51 121
30548 205 107 121 62 2 205
30554 1 22 17 61 119 1
30560 20 0 205 60 32 62
30566 22 215 62 15 215 62
30572 16 215 58 131 123 198
30578 48 215 201 22 9 0
30584 84 73 84 76 69 32
30590 111 102 32 80 65 71
30596 69 83 32 40 49 48
30602 32 67 104 114 115 46
30608 32 77 65 88 41 32
30614 63 62 2 205 1 22
30620 17 117 119 1 34 0
30626 205 60 32 201 22 11
30632 0 84 73 84 76 69
30638 32 111 102 32 80 65
30644 71 69 83 32 116 111
30650 32 98 101 32 76 79
30656 65 68 69 68 6 34
30662 63 34 32 68 65 84
30668 65 32 88 36 40 41
30674 62 2 205 1 22 17
30680 166 119 1 44 0 205
30686 60 32 201 22 15 0
30692 83 119 105 116 99 104
30698 32 111 110 32 116 97
30704 112 101 32 38 32 112
30710 114 101 115 115 32 69
30716 78 84 69 82 62 2
30722 205 1 22 17 225 119
30728 1 31 0 205 60 32
30734 201 22 17 7 79 46
30740 107 46 32 80 82 69
30746 83 83 32 65 78 89
30752 32 75 69 89 62 2
30758 205 1 22 17 15 120
30764 1 21 0 205 60 32
30770 205 101 120 201 22 5
30776 14 77 69 78 85 22
30782 7 5 49 41 32 39
30788 78 79 82 77 65 76
30794 39 22 9 5 50 41
30800 32 39 78 69 87 39
30806 22 11 5 51 41 32
30812 32 82 69 68 69 70
30818 73 78 69 42 176 92
30824 34 54 92 201 0 0
30830 0 62 2 205 1 22
30836 6 24 205 68 14 17
30842 54 120 1 47 0 205
30848 60 32 0 205 101 120
30854 201 22 0 14 77 69
30860 78 85 22 2 5 49
30866 41 32 83 84 65 82
30872 84 22 4 5 50 41
30878 32 83 84 79 80 22
30884 6 5 51 41 32 69
30890 82 65 83 69 32 65
30896 32 80 65 71 69 22
30902 8 5 52 41 32 83
30908 65 86 69 32 80 65
30914 71 69 83 22 10 5
30920 53 41 32 76 79 65
30926 68 32 80 65 71 69
30932 83 22 12 5 54 41
30938 32 67 72 65 78 71
30944 69 32 84 89 80 69
30950 70 65 67 69 22 20
30956 5 127 32 49 57 56
30962 50 32 83 116 117 97
30968 114 116 32 76 105 99
30974 104 111 108 108 115 22
30980 16 14 95 95 95 95
30986 205 107 121 62 2 205
30992 1 22 6 24 205 68
30998 14 17 135 120 1 131
31004 0 205 60 32 0 0
31010 0 0 201 62 120 50
31016 72 92 33 141 92 203
31022 246 205 61 127 201 58
31028 8 92 214 48 50 131
31034 123 195 136 123 58 8
31040 92 254 110 192 58 131
31046 123 60 254 5 32 2
31052 62 1 50 131 123 205
31058 89 123 33 1 1 34
31064 139 127 201 121 91 117
31070 93 112 127 97 126 115
31076 124 100 92 102 123 103
31082 125 42 54 92 34 176
31088 92 33 0 60 34 54
31094 92 201 0 205 80 127
31100 58 8 92 50 245 126
31106 254 13 32 4 205 141
31112 127 201 58 139 127 254
31118 43 40 41 58 245 126
31124 254 32 32 4 205 153
31130 127 201 214 32 56 26
31136 214 96 208 58 145 92
31142 203 135 50 145 92 205
31148 244 126 58 145 92 203
31154 199 50 145 92 205 146
31160 127 201 58 245 126 254
31166 12 32 4 205 84 122
31172 201 205 180 127 201 58
31178 139 127 254 43 200 58
31184 8 92 33 91 121 1
31190 16 0 237 177 120 177
31196 40 6 126 50 245 126
31202 24 191 58 8 92 254
31208 122 192 205 172 14 201
31214 58 140 127 50 125 122
31220 205 118 122 24 13 62
31226 22 245 50 125 122 205
31232 118 122 241 61 32 245
31238 205 89 123 201 205 112
31244 123 229 175 237 82 237
31250 75 139 127 17 42 0
31256 25 16 253 235 225 237
31262 82 235 122 179 200 35
31268 54 32 27 122 179 32
31274 248 42 139 127 62 23
31280 148 71 79 12 197 205
31286 0 14 193 13 32 248
31292 201 205 112 123 175 237
31298 82 237 75 139 127 17
31304 42 0 25 16 253 237
31310 82 35 13 32 252 201
31316 42 139 127 45 32 8
31322 46 42 37 32 3 33
31328 1 1 34 139 127 205
31334 61 122 54 32 58 145
31340 92 203 135 50 145 92
31346 205 88 127 201 205 112
31352 123 175 237 82 6 14
31358 17 42 0 25 16 253
31364 34 8 123 237 82 35
31370 34 10 123 1 0 42
31376 126 254 32 32 5 12
31382 35 16 247 201 120 6
31388 0 237 91 10 123 79
31394 237 176 42 8 123 71
31400 62 42 144 40 6 71
31406 54 32 43 16 251 24
31412 0 42 10 123 6 41
31418 35 126 254 32 40 8
31424 16 248 35 126 254 32
31430 32 4 16 248 24 56
31436 42 8 123 126 254 32
31442 32 48 42 10 123 1
31448 42 0 13 40 247 35
31454 126 254 32 40 247 43
31460 126 254 32 32 22 197
31466 229 42 8 123 229 209
31472 43 237 184 42 8 123
31478 126 254 32 32 7 225
31484 193 13 35 35 24 216
31490 225 225 201 0 0 0
31496 77 102 36 102 205 104
31502 123 229 175 237 82 17
31508 42 0 1 1 1 25
31514 16 253 237 62 35 13
31520 32 252 209 213 229 235
31526 237 82 17 1 0 35
31532 237 82 40 2 48 3
31538 225 225 201 229 193 24
31544 0 209 42 41 123 25
31550 237 176 225 24 13 225
31556 209 43 229 9 237 184
31562 225 237 91 41 123 25
31568 58 41 123 54 32 43
31574 61 32 250 6 24 197
31580 6 23 205 0 14 193
31586 16 247 205 148 123 201
31592 205 80 127 205 112 123
31598 201 0 42 75 92 126
31604 254 216 40 3 35 24
31610 248 17 7 0 25 17
31616 156 3 6 1 25 16
31622 253 201 205 112 123 54
31628 32 43 27 122 179 32
31634 248 201 205 112 123 6
31640 22 14 42 126 50 245
31646 126 237 67 26 127 229
31652 197 205 244 126 193 225
31658 43 13 32 237 16 233
31664 33 40 0 17 15 0
31670 229 205 181 3 225 17
31676 1 0 167 237 82 32
31682 240 201 6 50 197 33
31688 0 1 17 1 0 229
31694 205 181 3 225 17 16
31700 0 167 237 82 32 240
31706 193 16 233 201 0 0
31712 0 0 0 0 0 0
31718 0 0 16 16 16 16
31724 0 16 0 0 20 20
31730 0 0 0 0 0 0
31736 0 40 124 40 124 40
31742 0 0 16 56 80 56
31748 20 120 16 0 0 100
31754 104 16 44 76 0 0
31760 32 80 32 84 72 52
31766 0 0 16 16 0 0
31772 0 0 0 0 8 16
31778 16 16 16 8 0 0
31784 32 16 16 16 16 32
31790 0 0 0 40 16 124
31796 16 40 0 0 0 16
31802 16 124 16 16 0 0
31808 0 0 0 0 8 8
31814 16 0 0 0 0 124
31820 0 0 0 0 0 0
31826 0 0 16 16 0 0
31832 0 4 8 16 32 64
31838 0 0 56 76 84 84
31844 100 56 0 0 12 13
31850 24 12 45 123 12 123
31856 56 68 8 16 32 124
31862 0 0 120 4 56 4
31868 4 120 0 0 64 72
31874 72 124 8 8 0 0
31880 124 64 120 4 4 120
```

```
31886  0    0    56   64   120  68
31892  68   56   0    0    124  4
31898  4    8    16   32   0    0
31904  56   68   56   68   68   56
31910  0    0    56   68   68   68
31916  4    56   0    0    0    16
31922  16   0    16   16   0    0
31928  0    16   16   0    16   16
31934  32   0    0    16   32   64
31940  32   16   0    0    0    0
31946  124  0    124  0    0    0
31952  0    16   8    4    8    16
31958  0    0    56   68   8    16
31964  0    16   0    0    56   76
31970  84   92   64   56   0    0
31976  56   68   68   124  68   68
31982  0    0    120  68   120  68
31988  68   120  0    0    56   68
31994  64   64   68   56   0    0
32000  120  68   68   68   68   120
32006  0    0    124  64   120  64
32012  64   124  0    0    124  64
32018  120  64   64   64   0    0
32024  56   68   64   92   68   56
32030  0    0    68   68   124  68
32036  68   68   0    0    56   16
32042  16   16   16   56   0    0
32048  28   8    8    8    72   48
32054  0    0    72   80   96   80
32060  72   68   0    0    64   64
32066  64   64   64   124  0    0
32072  68   108  84   68   68   68
32078  0    0    68   100  84   76
32084  68   68   0    0    56   68
32090  68   68   68   56   0    0
32096  120  68   68   120  64   64
32102  0    0    56   68   68   68
32108  84   56   4    0    120  68
32114  68   120  72   68   0    0
32120  56   64   56   4    68   56
32126  0    0    124  16   16   16
32132  16   16   0    0    68   68
32138  68   68   68   56   0    0
32144  68   68   68   68   40   16
32150  0    0    68   68   84   84
32156  108  40   0    0    68   40
32162  16   16   40   68   0    0
32168  68   40   16   16   16   16
32174  0    0    124  8    16   32
```

```
32180  64   124  0    0    48   32
32186  32   32   32   48   0    0
32192  0    64   32   16   8    4
32198  0    0    24   8    8    8
32204  8    24   0    0    16   56
32210  84   16   16   16   0    0
32216  0    0    0    0    0    0
32222  0    252  24   36   32   112
32228  32   32   124  0    0    56
32234  4    60   68   60   0    0
32240  64   120  68   68   68   120
32246  0    0    0    60   64   64
32252  64   60   0    0    4    60
32258  68   68   68   60   0    0
32264  0    56   68   120  64   56
32270  0    0    24   36   32   120
32276  32   32   0    0    0    56
32282  68   68   68   60   4    56
32288  64   120  68   68   68   68
32294  0    0    16   0    48   16
32300  16   56   0    0    16   0
32306  48   16   16   16   80   32
32312  64   64   72   112  72   68
32318  0    0    64   64   64   64
32324  68   56   0    0    0    40
32330  84   84   84   84   0    0
32336  0    120  68   68   68   68
32342  0    0    0    56   68   68
32348  68   56   0    0    0    120
32354  68   68   68   120  64   64
32360  0    60   68   68   68   60
32366  4    4    0    56   68   64
32372  64   64   0    0    0    56
32378  64   56   4    120  0    0
32384  32   120  32   32   36   24
32390  0    0    0    68   68   68
32396  68   60   0    0    0    68
32402  68   68   40   16   0    0
32408  0    84   84   84   84   40
32414  0    0    0    68   40   16
32420  40   68   0    0    0    68
32426  68   68   68   60   4    56
32432  0    124  8    16   32   124
32438  0    0    48   32   32   64
32444  32   32   48   0    16   16
32450  16   16   16   16   0    0
32456  24   8    8    4    8    8
32462  24   0    0    0    52   88
32468  0    0    0    0    112  136
```

```
32474  232  200  232  136  112  0
32480  0    56   8    92   214  48
32486  195  78   121  0    17   1
32492  0    33   50   0    205  181
32498  3    201  33   32   0    41
32504  41   41   237  91   54   92
32510  25   6    8    14   6    86
32516  123  61   32   2    203  34
32522  203  34   220  22   127  13
32528  32   248  35   16   236  201
32534  197  213  229  33   1    1
32540  45   175  125  7    7    133
32546  133  111  203  35   203  35
32552  203  36   62   175  148  103
32558  62   6    145  133  79   120
32564  132  71   205  229  34   225
32570  209  193  201  33   1    60
32576  34   54   92   201  33   224
32582  122  34   54   92   201  0
32588  0    0    0    0    58   145
32594  92   203  199  50   145  92
32600  175  58   140  127  7    7
32606  7    71   62   183  144  71
32612  22   8    58   139  127  61
32618  79   7    7    129  129  79
32624  30   6    213  197  205  229
32630  34   193  209  12   29   32
32636  245  5    21   32   229  42
32642  139  127  34   26   127  34
32648  23   123  201  2    1    42
32654  139  127  24   16   205  61
32660  122  58   245  126  119  42
32666  139  127  44   125  254  43
32672  32   11   46   1    36   [illegible]
32678  254  23   32   3    33   43
32684  22   34   139  127  205  [illegible]
32690  126  201  1    254  254  237
32696  120  254  254  192  42   139
32702  127  1    254  247  237  120
32708  254  239  32   6    45   32
32714  46   44   32   43   1    254
32720  239  237  120  254  251  32
32726  13   44   125  254  44   40
32732  4    254  43   32   24   45
32738  24   21   254  247  32   6
32744  37   32   14   36   32   11
32750  254  239  32   10   36   124
32756  254  23   32   1    37   34
32762  139  127  205  234  126  201
```

GRÁFICOS 3D

SPECTRUM 16/48 K

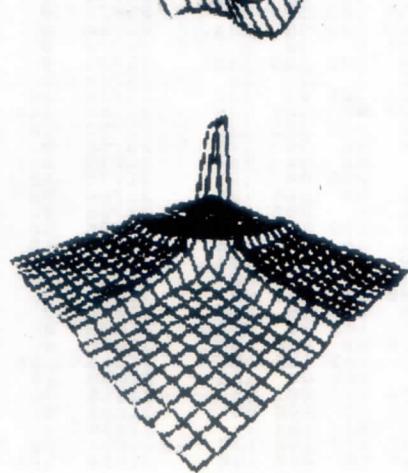
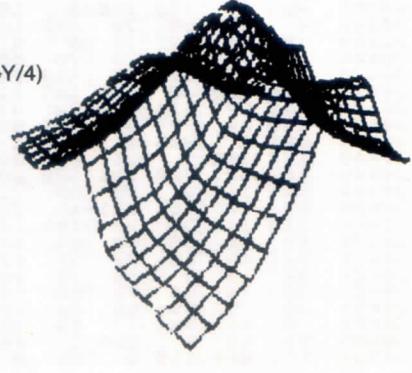
Autor.: Manuel Quinaz
Porto

Após a entrada do programa, ser-lhe-á pedido Z = função (x,y), seguindo-se os valores iniciais e finais para x e y. É então traçado o gráfico em 3 dimensões e em perspectiva dessa função.

EXEMPLOS:

SEN X + COS Y
X varia entre 0 e 8
Y varia entre 0 e 8

1/(ABS(x*Y))* . 5
X varia entre −3 e 3
Y varia entre −3 e 3

SEN X
X varia entre −8 e 8
Y varia entre 0 e 1

EXP (−X*X) + EXP (−Y*Y/4)
X varia entre −4 e 4
Y varia entre −4 e 4

```
   0> ***************************
      *   Programa elaborado    *
      *          por  :         *
      *  Manuel Jose Quinaz     *
      *   Em 6/7/1984           *
      ***************************
  10 BORDER 1: PAPER 1: INK 7: C
LEAR : PAPER 1: INK 7
  40 PRINT AT 20,3;"Espere um mo
mento";AT 13,8; FLASH 1;" PARE O
 GRAVADOR ": DIM x(21,21): FOR n
=0 TO 20: FOR m=0 TO 20: LET x(m
+1,n+1)=(45568-850*m+5202*n)/(35
6+17*(m+n)): NEXT m: NEXT n: GO
SUB 9910
  50 CLS : PRINT "  Este e um pr
ograma que permite a representac
ao tridimensional  e em prespect
iva de funcoes de  duas variavei
s.": PRINT : PRINT "  Deve sempr
e denominar as vari-aveis indepe
ndentes ""x"" e ""y"""
  60 PRINT AT 21,3;" PRIMA UMA T
ECLA "
  70 IF INKEY$="" THEN GO TO 70
  80 CLS
  90 INPUT "Introduza z=f(x,y) "
; LINE a$
 100 DEF FN f()=VAL a$
 110 DIM z(22,22)
 120 INPUT "Valor inicial de x="
;xi: INPUT "Valor final de x=";x
f
 130 IF xi=xf THEN PRINT FLASH 1
;"ERRO:O valor inicial e igual a
o      valor final.": GO TO 120
 140 INPUT "Valor inicial de y="
;yi: INPUT "Valor final de y=";y
f
 150 IF yi=yf THEN PRINT FLASH 1
;"ERRO:O valor inicial e igual a
o      valor final.": GO TO 140
 170 POKE 23674,0: POKE 23673,0:
 POKE 23672,0
 180 LET m=1: LET n=1: LET x=xi:
 LET y=yi: LET zi=FN f(): LET zf
=FN f()
 190 FOR x=xi TO xf STEP (xf-xi)
/20.9999: FOR y=yi TO yf STEP (y
f-yi)/20.9999
 200 IF m=2 AND n=1 THEN GO SUB
9800
 205 LET z(m,n)=FN f()*(x(m,n)+5
0)/(178+17*(m+n-2))
 210 IF z(m,n)>zf THEN LET zf=z(
m,n)
 220 IF z(m,n)<zi THEN LET zi=z(
m,n)
 230 LET n=n+1: NEXT y: LET n=1:
 LET m=m+1: NEXT x
 240 LET a=55/(zf-zi)
 300 CLS : FOR m=1 TO 21: FOR n=
1 TO 21
 310 LET x=x(m,n): LET y=178*(17
*m-128+x(m,n))/(178+17*m)+a*(z(m
,n)-zi): PLOT x,y: IF n>1 THEN D
RAW x(m,n-1)-x,178*(17*m-128+x(m
,n-1))/(178+17*m)+a*(z(m,n-1)-zi
)-y: PLOT x,y
 320 IF m>1 THEN DRAW x(m-1,n)-x
,178*(17*(m-1)-128+x(m-1,n))/(17
8+17*(m-1))+a*(z(m-1,n)-zi)-y
 340 NEXT n: NEXT m
 400 INPUT "Parar ?: (s/n) ";r$
 410 IF r$="s" OR r$="S" THEN ST
OP
 420 IF r$<>"n" AND r$<>"N" THEN
 GO TO 400
 430 INPUT "Nova funcao?: (s/n)
";r$
 440 IF r$="s" OR r$="S" THEN GO
 TO 90
 450 IF r$="n" OR r$="N" THEN GO
 TO 120
 460 GO TO 430
9810 LET seg=INT (.4*PEEK 23672+
102.4*PEEK 23673+26214.4*PEEK 23
674-5)
9820 LET min=INT (seg/60): LET s
eg=seg-60*min: LET hor=INT (min/
60): LET min=min-60*hor
9830 PRINT AT 0,2;"Demorara apro
ximadamente"
9840 PRINT AT 2,2;hor;" Horas  "
;min;" min.  ";seg;" seg."
9850 RETURN
9910 FOR n=-20 TO 30: BEEP .01,n
: NEXT n: RETURN
```

COSMAZOIGS

SPECTRUM 16/48

Baseado no jogo Asteróides, apenas com algumas alterações gráficas.

O JOGO:

O Objectivo do jogo é destruir todos os objectos que possam colidir com a nave que controlas. Podes rodar a nave para qualquer direcção (teclas 6 e 7) ou mesmo perseguir os cosmazoigs (tecla 9), para disparar (tecla 0) e a (tecla 8) para parar o jogo.

Para esta tua missão tens apenas 3 naves, por isso escolhe o nível de jogo que preferes, (de 0 a 9).

Se conseguires o record, escreve o teu nome e a seguir "ENTER", caso contrário, regressarás à 1.ª letra.

O PROGRAMA:

Retirado da revista "YOUR COMPUTER", Mar/84.

É necessário introduzir 2 programas para conseguires o jogo:

1 — (listagem 1) que deves gravar com o comando: SAVE "Cosmazoigs" LINE 1 e verificar depois de gravado (VERIFY"").

2 — (listagem 2) introduz o código máquina com o máximo de atenção para evitares erros. Depois de passado, para verificares se tudo está em condições, far "RUN". Se ocorreu algum erro, o computador indicar-te-à a sua localização.

Se tudo está perfeito, obterás a informação: "PODES GRAVAR". Em último caso, para verificares se realmente tudo está correcto, podes fazer:

LET L = USR 30448 : BORDER 0

Se houver incorrecção o programa "Aborta".

Listagem 1

```
  20 BORDER 0: PAPER 0: INK 7
  30 BRIGHT 1: CLEAR 30440
  35 POKE 23676,127
  40 PRINT AT 11,4; FLASH 1;"""C
OSMAZOIGS""A carregar!"
  60 PRINT AT 9,10; INVERSE 1; F
LASH 1;"espere!"
  70 FOR F=12 TO 15: BEEP .05,F
  80 NEXT F
  90 INK 0: PRINT AT 15,0;
  92 LOAD ""CODE
  95 INK 7: PAPER 0: CLS
```

```
  96 PRINT AT 10,11;"COSMAZOIGS"
  97 GO SUB 2300
  98 DIM N$(11,22)
  99 FOR F=1 TO 11
 105 LET N$(F)="00000_____________
______": NEXT F
 107 CLS
 110 PRINT AT 6,11;"OS PONTOS:"
 120 PRINT AT 8,8; INK 2;CHR$ 14
5; INK 7;"......10 pontos"
 130 PRINT AT 10,8; INK 4;CHR$ 1
46; INK 7;".....100 pontos"
 140 PRINT AT 12,8; INK 6;CHR$ 1
47; INK 7;"....1000 pontos"
 145 PAUSE 100: CLS
 150 PRINT AT 4,9;"USE AS TECLAS
:"
 160 PRINT AT 7,8;"6> Rodar esqu
erda"
 170 PRINT AT 9,8;"7> Rodar dire
ita"
 180 PRINT AT 11,8;"8> Parar o j
ogo"
 190 PRINT AT 13,8;"9> Avancar"
 200 PRINT AT 15,8;"0> Fogo"
 210 PAUSE 100: CLS
 220 PRINT TAB 11;"RECORDS:"
 230 PRINT
 240 FOR F=1 TO 10
 245 PRINT 'TAB 5; INK 6-INT ((F
-1)/2);N$(F, TO 5); INK 7;N$(F,6
 TO )
 250 NEXT F
 260 PAUSE 100: CLS
 270 IF INKEY$="" THEN GO TO 107
 280 CLS : BEEP .2,10
 285 IF INKEY$<>"" THEN GO TO 28
5
 290 INPUT TAB 7; BRIGHT 1;"Nive
l ?(0 TO 9)"; LINE A$
 300 IF LEN A$<>1 THEN GO TO 290
 305 IF A$(1)<"0" OR A$(1)>"9" T
HEN GO TO 290
 310 LET A=13-VAL A$
 320 POKE 32767,A
 330 PRINT AT 0,0; INK 5;"Score=
"; INK 7;"00000 "; INK 3;CHR$ 15
2;CHR$ 152;CHR$ 152; INK 5;"high
 score="; INK 7;N$(1, TO 5)
 340 LET L=2
 350 LET x=USR 30448
 360 PRINT AT 0,12+L;" "
 370 LET x=USR 30470
 380 LET L=L-1
 390 IF L<>0 THEN GO TO 360
 400 BORDER 0
 410 LET X$=""
 420 FOR F=6 TO 10
 430 LET X$=X$+STR$ (PEEK (F+317
44)-144)
 440 NEXT F
 450 LET F=1
 460 IF F=11 THEN GO TO 670
 470 IF X$<n$(F, TO 5) THEN LET
F=F+1: GO TO 460
 480 FOR G=10 TO F STEP -1
 490 LET N$(G+1)=N$(G)
 500 NEXT G
 510 LET N$(F)=X$+"
 "
 520 FOR G=1 TO 21
 530 PRINT AT G,0;"
 "
 540 NEXT G
 545 FOR G=232322 TO 23295
 546 POKE G,0: NEXT G
 550 PRINT AT 7,5; INK 4;"o melh
or de hoje";AT 9,1;"score maximo
 escreva o seu";AT 11,14;"Nome";
AT 14,5; INK 6-INT ((F-1)/2);X$;
 INK 7;"________________"
 560 LET G=11: POKE 23658,0
 570 LET A$=INKEY$
 580 IF A$="" THEN GO TO 570
 590 IF CODE A$=13 THEN GO TO 68
0
 600 IF A$<" " OR A$>"z" THEN GO
 TO 570
 610 PRINT AT 14,G;A$
 620 LET N$(F,G-4)=A$
 630 BEEP .01,20: LET G=g+1
 640 IF G=27 THEN LET G=11
 650 IF INKEY$<>"" THEN GO TO 65
0
 660 GO TO 570
 670 PRINT AT 11,11; FLASH 1;"Ac
abou o jogo"
 680 FOR F=1 TO 100: NEXT F
 690 GO TO 107
 999 STOP
1010 DATA 5,6,6,1.25,1,.75,8,6.5
.1,.5,6,T,8,T,10,T,99
1020 DATA 8,1,5,1.5,1,.5,6,T,8,T
.10,T,8,2.5,1,.5,6,T,8,T,10,T,8,
1,5,1.5,5,.5,6,T,5,T,1,T,1,2.5,9
9
1030 DATA 1,.5,6,T,8,T,10,T,99
1040 DATA 13,.5,13,T,10,T,8,T,99
1050 DATA 12,.75,8,.25,10,.75,6,
.25,8,.75,13,.25,12,T,10,T,8,T,1
2,1,6,1,5,.5,13,.5,12,T,10,T,8,T
.99
1060 DATA 12,.75,8,.25,10,.75,6,
.25,8,.75,5,.25,6,T,5,T,1,T,1,2.
5,13,.5,12,T,10,T,8,T,99
1070 DATA 12,1,10,.75,13,.25,8,.
75,5,.25,6,T,5,T,1,T,1,2.5,1,.5,
6,T,8,T,10,T,99
1080 DATA 12,.75,8,.24,10,.75,6,
.25,8,.75,5,.25,6,T,5,T,1,T,1,2.
5,1,.5,6,T,8,T,10,T,99
1090 DATA 8,1,5,1.5,1,.5,13,T,12
.T,10,T,8,2.5,1,.5,13,T,12,T,10,
T,8,1,5,1.5,5,.5,6,.5,8,.5,13,2.
5,1,.5,6,T,5,T,1,T,5,2.5,1,.5,6,
T,5,T,1,T,1,2.5,1,.5,6,T,5,T,1,T
.1,2.5,1,.5,6,T,5,T,1,T,1,3,-4,1
.1,3,-4,1,-11,4,99
2000 LET T=1/3
2005 READ N
2010 IF N=99 THEN RETURN
2020 READ M
2030 BEEP M/5,N
2040 GO TO 2005
2110 RESTORE 1020: GO SUB 2000
2120 GO SUB 2000
2130 RESTORE 1020: GO SUB 2000
2140 RESTORE 1040: GO SUB 2000
2150 GO SUB 2000
2160 GO SUB 2000
2170 RESTORE 1050: GO SUB 2000
2180 RETURN
2210 FOR z=1 TO 12
2220 BEEP 1/8,-11
2230 NEXT z
2240 BEEP .2,1
2250 BEEP .8,8
2260 RETURN
2310 GO SUB 2200: GO SUB 2200
2320 GO SUB 2200
2330 FOR z=1 TO 16
2340 BEEP 1/8,-11
2350 NEXT Z
2360 RESTORE 1000: GO SUB 2000
2370 GO SUB 2100
2380 RESTORE 1070: GO SUB 2000
2390 GO SUB 2100
2400 RESTORE 1080: GO SUB 2000
2410 GO SUB 2000
2420 RETURN
```

Listagem 2

```
   5 CLEAR 30440
  10 BORDER 0: INK 3: PAPER 0
  15 CLS : OVER 1
  20 PLOT 128,0
  25 DRAW 0,165,2771*PI
  30 PLOT 128,0
  35 DRAW 0,165,-2771*PI
  40 OVER 0: INK 7: BRIGHT 1
  45 PRINT AT 0,2;"""COSMAZOIGS"
 AdP. CLUBE Z80"
  50 PRINT AT 11,11; FLASH 1;"Um
momento!"
  55 GO TO 500
  90 DATA "7F582466E73838E766240
```

```
82A1C7F7F1C2A083C4E9FBFFFDF7E3C3
C66FFA5A5FF663C00000018180000000
00000181800000000000018180000000
000001818000000018182424425AA5C30
30D32C23414080808C0B04C23234CB0C00
8081434C2320D03C3AD5A42242418181
010282C434CB0C0030D32C4C4320D03C
0B04C432C2810100"
 110 DATA "7F00C5D5E577CB7F200E4
FC6405F167F1A4711587F791807E67F0
64711003DE56F2600292929 19D1EB7CD
6246770D6508767876706081A7713241
0FAE1D1C1C93"
 130 DATA "7AD04210FE3E10A94FD3F
E7A8357C9F3C53A485C0F0F0F4FCB7B2
807CDD07A38FB1805CDD07A30FBC1FBC
90"
 150 DATA "7AA07CFE7F2009267C7DE
61FC6206FC9D511E083EB19EBD1D87DE
61FC6E06F267EC92"
 170 DATA "7A70FFE0FFE1000100210
020001FFFFFFFDFF5D5E60787C670E52
67A6F56235EE119CDA07AD1F1C98"
 190 DATA "7A5021207C3E80CD007FE
511107FEBED52110400CD8503E1237CF
E7F20E7C93"
 210 DATA "7A002AFE7F06017EFE04D
83E80CD007F01FEEFED403AFD7F4FCB6
0200010DCB58200010C79E6074FCB48CC8
07A7932FD7F22FE7F06017EFE04D8047
9C608C3007F2"
 230 DATA "7F4047424446474747474
747474747474747475"
 250 DATA "7FF4000A030100000000
001807D9"
 270 DATA "79D0DDE5E5D5C5267CC60
66F7E3CFE9A20083E90CD007F2818F2C
D007F1108002100002CD8503C1D1E1DDE
1C95"
 290 DATA "7980E521FB7F5E2356131
37AFE1020021600EB22FB7F5E2356E1D
5C1C92"
 310 DATA "7950CD507A21007B36802
37CFE7C20F8CDB0797AE60F260087876
7878730001246F01807C09CB7A20043E1
F856FEB2100783E0377237323372EBCD0
07F21907D22FE7FCDB0797BE60732037
6AF32FD7FC608C3007F7"
 330 DATA "7940DD6E01DD66027EF53
E80CD007FF1C94"
 350 DATA "78E0CD4079FE043804FE0
83817DD7E03CD807A3E03CD007FDD750
1DD7402110001DD19C9DDE5112000060
8783DDD7703E5CD807ADD7501DD74023
E02DD7700CD007FE1DD1910E43E01CDD
079DDE1110001DD19C90"
 370 DATA "7880CD4079FE043804FE0
83817DD7E03CD807A3E02CD007FDD750
1DD7402112000DD19C9DDE5110400060
8783DDD7703E5CD807ADD7501DD74023
E01DD7700CD007FE1DD1910E43E02CDD
079DDE1112000DD19C93"
 390 DATA "7850CD4079FE043804FE0
83813DD7E03CD807A3E01CD007FDD750
1DD7402180 9DD3600803E03CDD079110
400DD19C99"
 410 DATA "77E03AF77F3D201501FEE
FED780FD83AFD7F32FA7F2AFE7F22F87
F3E0A32F77F4F2AF87F7EFE043817FE0
830053E80CD007F3AFA7FCD807A22F87
F7EFE8028073E0132F77F18053AF77F3
DC83AFA7FE603C604CD007F79878787 8
7C680571E08C3DD7A9"
 430 DATA "77A00601DD21007BDD7E0
0FE802007110400DD19181904C53D200
5CD5078180E3D2005CD8078180 63D200
3CDE078C1DDE5E17CFE7C20D105C03E0
132F77FC350790"
 450 DATA "774021207C3E80CD007F2
37CFE7F20F5CDA077CDB0797AE6070F0
F0F267D6F7BE60FC60885 6F7EFE8020E
822FE7F3AFD7FE607C608CD007F11010
1CDDD7A0608783D87878732485C11018
0CDDD7A10F03E0132F77FC92"
 470 DATA "76F021067C3E90CD007F2
37DFE0B20F5CD50793E0132F77F3AF67
F3D200CCDA077CD007A05CA40773E033
2F67FCDE0772AF47F7CB52B20FB3A085
CFE3828F916D74"
 500 LET A=10: LET B=11
 510 LET C=12: LET D=13
 520 LET E=14: LET F=15
 525 LET T=0
 530 FOR L=90 TO 470 STEP 20
 540 RESTORE L
 550 READ A$
 560 FOR X=1 TO LEN A$-1
 570 LET T=T+VAL A$(X): NEXT X
 580 LET T$=STR$ T
 585 LET T$=T$(LEN T$)
 590 IF T$<>A$(LEN A$) THEN GO T
O 900
 600 LET Y=4096*VAL A$(1)
 610 LET Y=Y+256*VAL A$(2)
 620 LET Y=Y+16*VAL A$(3)
 630 LET Y=Y+VAL A$(4)
 640 IF Y<30440 THEN GO TO 900
 650 LET A$=A$(5 TO )
 660 LET X=16*VAL A$(1)+VAL A$(2
)
 670 POKE Y,X: LET Y=Y+1
 680 LET A$=A$(3 TO )
 690 IF A$<>T$ THEN GO TO 660
 700 NEXT L
 710 CLS
 720 PRINT AT 10,14;"PODE";AT 11
,13;"GRAVAR"
 730 CIRCLE 128,87,30
 740 STOP
 900 CLS
 910 PRINT AT 10,7; FLASH 1;"ERR
O NA LINHA";L
 920 FOR H=0 TO 2: BEEP .05,H
 930 BORDER H: NEXT H
 940 GO TO 920
```

PUZZLE DE PALAVRAS

SPE 16/48 K

Autor: PAULO CASTELO
PORTO

Retirado do "YOUR COMPUTER" Jan/84, este programa funciona tal como os puzzles, que vês publicados em várias revistas.

Dar entrada de 10 palavras chave que serão baralhadas num quadro em posições diversas: diagonal, vertical e horizontal. Nota que as palavras podem estar pela ordem inversa. Tens ainda um MENU com 5 opções.

A — Dá-te a resolução
C — Copia p/ a impressora
Q — Para
R — Recomeça
S — Baralha para um novo jogo

```
  10 PAPER 0: INK 7: BRIGHT 1: B
ORDER 0: CLS
  20 LET a$="        GERADOR DE PAL
AVRAS    ": FOR a=1 TO 31: PRINT A
T 0,0;a$(32-a TO ): BEEP .01,0:
NEXT a: LET A$="          ADAPTADO
 POR:    ": FOR A=1 TO 31: PRINT A
T 1,0;a$(32-a TO ): BEEP .01,0:
NEXT a
  30 LET A$="           CLUBE Z80
       ": FOR A=1 TO 31: PRINT
AT 3,0;A$(32-A TO ): BEEP .01,0:
 NEXT A
```

```
  40 FOR A=1 TO 1000: NEXT A: CL
S
  50 POKE 23658,255
  60 PRINT AT 0,0;
  70 PRINT TAB 6; PAPER 1;"Gerad
or de palavras"; PAPER 0;TAB 6;
INK 6;"▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀"
  90 PRINT : PRINT TAB 8;"qualqu
er tecla"
 100 IF INKEY$="" THEN GO TO 100
 110 CLS
 120 PRINT AT 20,0;"tecla enter.
"
 130 DIM a$(10,10)
 140 FOR a=1 TO 10
 150 INPUT "palavra ";(a);"= ";b
$: CLS : IF LEN b$>10 THEN PRINT
 "10 letras no maximo!": GO TO 1
50
 160 IF b$="" THEN GO TO 150
 170 LET a$(a)=b$: NEXT A: CLS
 180 PRINT AT 7,9; PAPER 1;"um m
omento"
 190 DIM g$(20,20): DIM c(20,20)
: DIM x(10): DIM y(10)
 200 FOR a=1 TO 10
 210 FOR b=1 TO 10: IF a$(a,b TO
 b)<>" " THEN NEXT b
 220 LET B=B-1
 230 LET X(A)=INT (RND*20)+1: LE
T Y(A)=INT (RND*20)+1
 240 LET d$="": LET xx=x(a): LET
 yy=y(a)
 250 LET d$=d$+("1" AND yy+b<21)
+("2" AND yy+b<21 AND xx>b)+("3"
 AND xx>b)+("4" AND xx>b AND yy>
b)+("5" AND yy>b)+("6" AND yy>b
AND xx+b<21)+("7" AND xx+b<21)+(
"8" AND xx+b<21 AND yy+b<21)
 260 LET d=VAL (d$(INT (RND*(LEN
 d$)+1)))
 270 FOR c=1 TO b: IF g$(xx,yy)=
" " THEN LET xx=xx-(d>1 AND d<5)
+(d>5 AND d<9): LET yy=yy-(d>3 A
ND d<7)+(d<3 OR d=8): NEXT c: GO
 TO 290
 280 GO TO 230
 290 LET ink=INT (RND*4)+4: LET
xx=x(a): LET yy=y(a): FOR c=1 TO
 b: LET g$(xx,yy)=a$(a,c): LET c
(xx,yy)=ink: LET xx=xx-(d>1 AND
d<5)+(d>5 AND d<9): LET yy=yy-(d
>3 AND d<7)+(d<3 OR d=8): NEXT c

 300 NEXT A: CLS
 310 FOR A=1 TO 20: FOR B=1 TO 2
0: IF G$(A,B)=" " THEN LET G$(A,
B)=CHR$ (64+(INT (RND*26)+1))
 320 NEXT B: NEXT A
 330 PLOT 174,174: DRAW -4,-4: P
LOT 174,1: DRAW -4,4: PLOT 1,1:
DRAW 4,4: PLOT 1,174: DRAW 4,-4:
 PLOT 5,5: DRAW 0,165: DRAW 165,
0: DRAW 0,-165: DRAW -165,0: PLO
T 1,1: DRAW 0,173: DRAW 173,0: D
RAW 0,-173: DRAW -173,0
 340 FOR A=1 TO 20: PRINT AT A,1
;G$(A): NEXT A
 350 FOR A=1 TO 10: PRINT AT A,2
2; INK 6;A$(A): NEXT A
 360 PRINT AT 13,22; PAPER 2;"Me
nu"
 370 PRINT AT 15,22;"A=respost."
 380 PRINT AT 17,22;"C=copia"
 390 PRINT AT 16,22;"Q=parar"
 400 PRINT AT 18,22;"R=recomeca"
 401 PRINT AT 19,22;"S=baralhar"
 410 IF INKEY$="C" THEN FOR A=14
 TO 19: PRINT AT A,22;"         "
 420 IF INKEY$="R" THEN RUN 120
 430 IF INKEY$="Q" THEN CLS : ST
OP
 435 IF INKEY$="S" THEN CLS : PR
INT AT 6,10; FLASH 1; PAPER 2;"B
ARALHANDO": GO TO 180
 440 IF INKEY$<>"A" THEN GO TO 4
10
 450 FOR A=20 TO 1 STEP -1: PRIN
T AT A,1; INK 2;G$(A): NEXT A
 460 FOR A=1 TO 20: FOR B=1 TO 2
0: IF C(B,A)>2 THEN PRINT AT B,A
; OVER 1; INK C(B,A);" "
 470 NEXT B: NEXT A
 480 PRINT #1;"Qualquer tecla -
Ecran normal": PAUSE 1: PAUSE 0:
 INPUT ""
 490 GO TO 340
```

ROTINA EM CÓDIGO MÁQUINA

(Resposta à pergunta de Mário Rebelo)

A Pergunta (publicada em Maio, pág. 7):

```
LDE,N
LDD,N
LDA,("AT"(22))N
RST16
LDA,D
RST16
LDA,E
RST16
LDA,"INR"(1E)N
RST16
LDA,N (car)
RST16
LDA,N (símbolo)
RST16
RET
```

"A rotina que listo ao lado, como podem ver, coloca em certa posição do écran um determinado símbolo com a côr que se pretende. O problema é que, após o simbolo, surge sempre um número que varia de linha para linha. Se souberem, gostaria que me explicassem porque é que isso acontece e como é possível eliminar tal número do écran".

MÁRIO REBELO/Coimbra

FERNANDO PRECES Responde:

"Quando se aplica o caracter 22 como comando "AT", numa rotina em C.M., sem que primeiro se abra o canal "S" (livre trânsito par o ecran), podem acontecer fenómenos muito estranhos.

É importante frisar que 2 computadores do mesmo modelo, ao depararem com uma instrução imprecisa, momentaneamente podem seguir caminhos diferentes. Assim no meu SPECTRUM 48K introduzi, na íntegra, a sua rotina máquina e nenhum símbolo adicional apareceu no écran. Em contrapartida, escreve o caracter sempre na linha 22, seja qual fôr o número de linha que eu proponha.

Para abrir o canal "S", tem dois procedimentos possíveis:

a) Programa em Basic, com pequenas rotinas em C.M.. Quando utilizar a sua rotina, escreva:

Linha n.º PRINT AT 0,0: RANDOMIZE USR X

b) Programa em C.M. utilize o seguinte Assembler:

```
LD A,2
CALL 5633     Para abertura do canal "S"
```

```
LD B,24
CALL 3652      Se pretender limpar o primeiro o ecran
LD A,22
RST 16
LD A,N
RST 16         Posição PRINT no ecran
L,D A,N
RST 16
LD A,16
RST 16
LD A,N         Cor "INK"
RST 16
LD A,17
RST 16
LD A,N         Cor "PAPER"
RST 16
LD A,N
RST 16         imprime o símbolo
RET
```

Para finalizar, devo acrescentar que esta rotina é muito interessante para estudo, mas tem pouca aplicação prática.
Se descodificar alguns programas comerciais existentes no mercado, encontrará rotinas para o efeito, bem mais rápidas e eficazes.

FERNANDO PRECES/Sacavém

SCROLL, BREAK, CONTINUE e VERIFY

RUI CARVALHO/Barreiro expõe "algumas anomalias do teclado do **Spectrum**, que gostaria de ver publicadas, e perguntas que gostaria de ver respondidas":

- Descobri que primindo CAPS SHIFT 3 quando nos é pedido "SCROLL?", este é executado duas vezes sem no-lo pedir — isto pode ser útil especialmente em listagens muito longas.
- A revista ZX COMPUTING (Ab/Maio 84) apresenta uma nova maneira de fazer "BREAK" num programa:

  Primir CAPS SHIFT, SYMBOL SHIFT e 2 ao mesmo tempo. No entanto, isto nem sempre funcionará. Por exemplo, quando aparece "SCROLL?", caso se prima estas 3 teclas ou outras, nada acontecerá; primindo CAPS SHIFT e SYMBOL SHIFT ao mesmo tempo, aparecerão várias coisas que não nos interessam pois a sua aplicação prática é nula.
  Note-se que se se quiser parar o "SCROLL?" a seguir a primir CAPS SHIFT, SYMBOL SHIFT e outra tecla, o ecran subirá (sendo portanto executado o "SCROLL?").
  Mas agora, só BREAK parará o "SCROLL?".
  Existem também outros casos em que esta combinação não funciona — p. ex.: PAUSE 0 (zero).
- Quando a "SCROLL?" respondemos N, seguido de CONT, porque é que o computador repete ou continua o comando CONTINUE? Porque é que "LIST", no mesmo caso, dá 0 (zero) OR?
- Por estranho que pareça, VERIFY não pode entrar na constituição de um programa, pois dá erro R: tape loading error (quando o gravador está a funcionar). Existe alguma maneira de VERIFY ser aceite pelo computador?".

RUI CARVALHO espera que os sócios discutam ou esclareçam estas questões. Escrevam-nas ao CLUBE Z80 para publicação.

***

VERIFIQUE A CAPACIDADE DO SEU ZX SPECTRUM

Publicamos no mês passado (n.º 21, pág. 6) uma indicação para verificar a capacidade do SPECTRUM: PRINT PEEK 23733 ENTER (se 255-48K; se 127-16K).

CARLOS OLIVEIRA/Portimão escreveu-nos o seguinte:

"Julgo ser mais agradável ver no ecran da TV 16 ou 48. Por isso, sugiro que se utilize a instrução PRINT PEEK 23733/4--15.75 ENTER".

***

Sobre a Rubrica "INTRODUÇÃO À LINGUAGEM MÁQUINA"

JORGE LANDECK/Seia sugere o seguinte:

"Tal como eu, penso que também outros sócios não tiveram a oportunidade de se tornarem tal, senão recentemente. Assim, alguns de nós apenas possuímos as revistas mais recentes o que, constitui, pelo menos para mim, uma grande lacuna (. . .). Venho pois sugerir que se faça, por exemplo a edição de uma revista especial, onde se resumisse a rubrica "Introdução à Linguagem Máquina" que penso de especial importância".

— Que pensam os outros sócios desta ideia? Escrevam-nos.

DUAS "PEQUENAS" ROTINAS

SPECTRUM

BEEP NO SPECTRUM

Talvez muitos dos possuidores do Spectrum estejam desapontados com o comando BEEP.
Aqui apresentamos uma pequena rotina em BASIC que lhe dará novas perspectivas sobre este comando:

```
  10 READ b: RESTORE
  20 FOR i=1 TO 32
  30 READ a
  40 BEEP 0.2,a: BEEP .05,a
  50 LET b=a
  60 NEXT i
  70 DATA 13,11,13,9,4,9,1,1
  80 DATA 13,11,13,9,4,9,1,1
  90 DATA 13,15,16,15,16,13,15,1
3
 100 DATA 15,11,13,11,13,15,16,1
6
```

NÃO PERMITIR O **MERGE**

Qualquer programa em BASIC pode ser parado, usando a instrucção MERGE. Para evitar a leitura do seu programa, apresentamos a seguir duas linhas que devem ser adicionadas à listagem e aproveitadas para a gravação do mesmo (GOTO 9998).
Desta forma será impossível fazer MERGE ou BREAK, ficando o programa a entrar com LOAD""CODE.

```
9998 POKE 23613,82: POKE 23614,6
4: SAVE "(nome do programa)" COD
E 23552,(valor aprox.)+1000

9999 GO TO (inicio do programa)

Para achar o comprimento aprox.
Faca: PRINT 41472-(65535-USR 796
2)
```

TOPS EM INGLATERRA

OS MICROS E PROGRAMAS (Spectrum) MAIS VENDIDOS, EM INGLATERRA, NA PRIMEIRA SEMANA DE JULHO (informações da revista PERSONAL COMPUTER NEWS, 14 Julho, 1984, n.º 69)

- **MICROS**

Preço até £: 1 000	
1 - Spectrum	£ 99
2 - CBM 64	£ 199
3 - Electron	£ 199
4 - Vic 20	£ 145
5 - BBC B	£ 399
6 - Oric Atmos	£ 175
7 - Memotech 500	£ 275
8 - Atari 800XL	£ 250
9 - ORIC	£ 99
10 - Dragom	£ 150

Preço superior £: 1 000	
1 - IBMPC	£ 2 390
2 - Apricot	£ 1 760
3 - Apple III	£ 2 755
4 - Sirius	£ 2 525
5 - TS 1603	£ 2 640
6 - DEC Rainbow	£ 2 359
7 - Compaq	£ 1 960
8 - Wang Professional	£ 3 076
9 - Philips P2000 C	£ 1 484
10 - LSI Octópus	£ 1 760

- **PROGRAMAS SPECTRUM (Jogos)**

1 — SABRE WULF*
2 — TLL
3 — PSYTRON
4 — MUGSY
5 — FIGHTER PILOT*
6 — HULK
7 — JACK & B'STALK
8 — LORDS OF MIDNIGHT
9 — TRASHMAN*
10 — JET SET WILLY*
11 — CODE NAME MAT
12 — VALHALLA*
13 — NIGHT GUNNER*
14 — CAVELON
15 — SCUBA DIVE*
16 — CHUKKIE EGG*
17 — ATIC ATAC*
18 — ANTICS
19 — FOOTBALL MANAGER*
20 — CHEQUERED FLAG*

(* Programas disponíveis no CLUBE Z80)

OS 10 MAIS VENDIDOS NO CLUBE Z80

(JOGOS SPECTRUM)

1 — JET SET WILLY
2 — FIGHTER PILOT
3 — PHEENIX
4 — CHEQUERED FLAG
5 — NIGHT GUNNER
6 — HUNTER KILLER
7 — SIMULADOR DE VOO (Psion)
8 — SPACE SHUTTLE
9 — PINBALL
10 — ATIC ATACK

NOVOS PROGRAMAS SPECTRUM

JOGOS **Preço**

- BLACK PLANET (. . .48K) — Séc. XXI: Black Planet, o planeta pirata serve de base para elementos criminosos. Você é "Starmagon" de uma patrulha galática que tem como objectivo a destruição das vias espaciais desses elementos. Para isso, terá de conseguir as 7 peças-chave necessárias à destruição do BLACK PLANET. **400$00**
- FRED (48K) — O herói do jogo explora um labirinto cheio de monstros, com o objectivo de apanhar tesouros, dispondo de uma arma com 6 balas. **400$00**
- GRID RUN (. . .16/48K) — Um labirinto com 2 automóveis. Sendo condutor de um, percorra rapidamente todo o labirinto, evitando o seu perdeguidor. **400$00**
- MUNCHER (. . .16/48K) — Outra versão de "Pacman" com 9 níveis de dificuldade e possibilidade de um ou 2 jogadores. **400$00**
- PHEENIX (. . .16/48K) — Jogo de arcádia muito popular, com vários níveis de dificuldade e diferentes fases durante a invasão inimiga. O jogo termina com a destruição da nave-mãe que é protegida ainda por alguns invasores. **400$00**
- POGO (48K) — Percorra uma pirâmide de hexágonos que vão mudando de côr, aquando da sua passagem. Há perseguidores a impedir o êxito da sua missão. **400$00**
- TANKS (. . .16/48K) — O seu objectivo é duplo: Destuir os tanques inimigos sem os deixar avançar além da linha de fogo e procurar uma base para abastecimento de combustível e munições. Atenção! O caminho está minado. Por isso oriente-se pelas diferentes cores correspondentes aos vários detectores de minas. **400$00**
- TRASHMAN (48K) — A finalidade do jogo é apanhar o número de contentores do lixo indicado, transportando-os até ao carro e colocando-os no seu lugar. **400$00**
- UBOAT HUNT (. . .48K) — Versão traduzida do "Hunter Killer" (destruir submarinos inimigos). **400$00**
- ZOMBIES (. . .16/48K) — Jogo de arcádia do género "Invadens", com apresentação gráfica bastante variada e com diferentes fases aumentando o nível de dificuldade. **400$00**

DESCONTO DE 20% PARA SÓCIOS DO CLUBE-VENDAS NA SEDE OU À COBRANÇA

OS PROGRAMAS SEGURANÇA E CDU

Nos **novos programas** anunciados em Maio, publicámos SEGURANÇA E CDU que, apesar da nossa insistência com o seu editor/autor, ainda não nos foram remetidos.

Lamentamos este facto e apenas podemos pedir desculpa a todos os sócios que os requisitaram. Os seus pedidos continuam "na gaveta" e serão atendidos quando dispusermos dos referidos programas.